



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ

FACULTY OF INFORMATION TECHNOLOGY

ÚSTAV POČÍTAČOVÉ GRAFIKY A MULTIMÉDIÍ

DEPARTMENT OF COMPUTER GRAPHICS AND MULTIMEDIA

HRA SET PRO ANDROID

SET GAME FOR ANDROID

BAKALÁŘSKÁ PRÁCE

BACHELOR'S THESIS

AUTOR PRÁCE

AUTHOR

ANTONÍN VLACH

VEDOUCÍ PRÁCE

SUPERVISOR

prof. Ing. ADAM HEROUT, PhD.

BRNO 2018

Vysoké učení technické v Brně - Fakulta informačních technologií

Ústav počítačové grafiky a multimédií

Akademický rok 2017/2018

Zadání bakalářské práce

Řešitel: **Vlach Antonín**

Obor: Informační technologie

Téma: **Hra Set pro Android**
Set Game for Android

Kategorie: Uživatelská rozhraní

Pokyny:

1. Seznamte se s problematikou návrhu a vývoje mobilních aplikací; zaměřte se na platformu Android a vývoj her.
2. Vyhledejte a analyzujte existující hry s podobným zaměřením (logické, karetní, pro více hráčů).
3. Navrhněte a prototypujte způsob interakce s aplikací a jednotlivé prvky uživatelského rozhraní.
4. Navrhněte a implementujte řešenou aplikaci.
5. Testujte vytvořenou aplikaci na uživateli a iterativně ji vylepšujte.
6. Zhodnoťte dosažené výsledky a navrhněte možnosti pokračování; vytvořte plakátek a krátké video pro prezentování projektu.

Literatura:

- Steve Krug: Don't Make Me Think, Revisited: A Common Sense Approach to Web Usability, ISBN-13: 978-0321965516
- Android Developers: <https://developer.android.com/index.html>
- Susan M. Weinschenk: 100 věcí, které by měl každý designér vědět o lidech, Computer Press, Brno 2012

Pro udělení zápočtu za první semestr je požadováno:

- Body 1 až 3, značné rozpracování bodů 4 a 5.

Podrobné závazné pokyny pro vypracování bakalářské práce naleznete na adrese

<http://www.fit.vutbr.cz/info/szz/>

Technická zpráva bakalářské práce musí obsahovat formulaci cíle, charakteristiku současného stavu, teoretická a odborná východiska řešených problémů a specifikaci etap (20 až 30% celkového rozsahu technické zprávy).

Student odevzdá v jednom výtisku technickou zprávu a v elektronické podobě zdrojový text technické zprávy, úplnou programovou dokumentaci a zdrojové texty programů. Informace v elektronické podobě budou uloženy na standardním nepřepisovatelném paměťovém médiu (CD-R, DVD-R, apod.), které bude vloženo do písemné zprávy tak, aby nemohlo dojít k jeho ztrátě při běžné manipulaci.

Vedoucí: **Herout Adam, prof. Ing., Ph.D., UPGM FIT VUT**

Datum zadání: 1. listopadu 2017

Datum odevzdání: 16. května 2018

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
Fakulta informačních technologií
Ústav počítačové grafiky a multimédií
602 00 Brno, Božetěchova 2



doc. Dr. Ing. Jan Černocký
vedoucí ústavu

Abstrakt

Cílem práce bylo vytvořit adaptaci karetní hry Set pro zařízení s operačním systémem Android. Požadovaná aplikace měla být jednoduchá na ovládání. Dalším požadavkem bylo, aby začínající uživatel nemusel znát pravidla hry předem, ale mohl se je naučit právě používáním této aplikace.

Výsledkem je nativní aplikace pro Android s jednoduchým material designem. Její hlavní částí je hra na čas. V pochopení pravidel může novým uživatelům pomoci vysvětlení pravidel doplněné grafickými příklady nebo také výukový mód – tutoriál, při kterém lze stupňovat obtížnost a hráč si může kdykoli vyžádat nápovědu.

Dokončení jedné hry trvá zhruba 3–10 minut (podle úrovně hráče), proto je hra ideální pro ukrácení dlouhé chvíle (např. při čekání na hromadnou dopravu).

Abstract

The aim of this thesis was to create an adaptation of card game Set for Android devices. The application should be easy to use. Another purpose of the application is to help beginner users with understanding the game rules.

The outcome is a native Android application using simple material design. The main part is a time game. There is also explanation of rules with some graphical examples. Tutorial mode provides more difficulty levels helping beginners to master the game.

One game takes about 3–10 minutes (depending on players skills), therefore the game is good to make some boredom more pleasant (e.g. waiting for public transport).

Klíčová slova

Set, Triáda, Hra, Android

Keywords

Set, Tripple, Game, Android

Citace

VLACH, Antonín. *Hra Set pro Android*. Brno, 2018. Bakalářská práce. Vysoké učení technické v Brně, Fakulta informačních technologií. Vedoucí práce prof. Ing. Adam Herout, PhD.

Hra Set pro Android

Prohlášení

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně pod vedením prof. Adama Herouta. Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

.....

Antonín Vlach
14. května 2018

Poděkování

V první řadě děkuji prof. Adamu Heroutovi za jeho vedení, za nadšení pro věc i za mnohé věcné připomínky. Dále chci poděkovat Ing. Jakubu Špaňhelovi za umožnění provedení testů na všech momentálně dostupných fakultních zařízeních. V neposlední řadě bych rád vyjádřil vděčnost všem testerům za jejich čas i užitečnou zpětnou vazbu.

Obsah

| | | |
|----------|--|-----------|
| 1 | Úvod | 2 |
| 2 | Rozbor hry Set | 3 |
| 2.1 | Pravidla hry | 3 |
| 2.2 | Podobné existující aplikace | 5 |
| 3 | Vývoj aplikací pro Android | 9 |
| 3.1 | Android Studio | 9 |
| 3.2 | Tvorba grafického uživatelského rozhraní | 10 |
| 3.3 | Další aspekty vývoje aplikací | 11 |
| 4 | Návrh řešení | 14 |
| 4.1 | Uživatelské rozhraní | 14 |
| 4.2 | Herní logika | 18 |
| 4.3 | Ukládání dat | 20 |
| 5 | Implementace | 22 |
| 5.1 | Průběh vývoje | 22 |
| 5.2 | Ladění, testování a publikování aplikace | 28 |
| 5.3 | Účast na konferenci Excel@FIT | 31 |
| 6 | Závěr | 33 |
| | Literatura | 34 |

Kapitola 1

Úvod

Tato práce pojednává o tvorbě mobilní aplikace pro chytré telefony nebo jiná zařízení používající systém Android. Vytvořená aplikace je programovou adaptací logické karetní hry Set. Ta spočívá v hledání trojic karet podle určitých podmínek. Práce rozebírá všechny fáze vývoje mobilní aplikace od počáteční přípravy přes tvorbu návrhu a samotnou tvorbu aplikace až po testování a zveřejnění hotové aplikace. Zmíněné fáze vývoje probíhají v uvedeném pořadí, ale navzájem se prolínají.

Mírně pokročilý hráč zvládne jednu hru dohrát do deseti minut, proto je hra ideálním prostředkem pro ukrácení dlouhé chvíle např. při čekání na hromadnou dopravu. Hra Set rozvíjí paralelní myšlení a postřeh, neboť je třeba vnímat několik vlastností karet současně, přičemž cílem hry je dosáhnout co nejlepšího času. Mobilních aplikací pro Android vzniká spousta každý den. Také hra Set už dostala několik svých aplikací. Smyslem této práce není tvořit již dříve vytvořené, ale naopak vycházet z existujících aplikací a poučit se z nich – inspirovat se jejich pozitivními stránkami a vyhnout se těm negativním.

Mou osobní motivací k výběru tohoto tématu byl velký zájem o hru Set. Znáám ji už dlouho. Líbí se mi mnohé existující aplikace, ale na každé z existujících aplikací mi také vadí některé aspekty. Proto jsem chtěl vytvořit vlastní aplikaci s co nejlepším uživatelským prostředím. Také ze svého okolí, od svých přátel, jsem zaznamenal zájem o takovou aplikaci.

Cílem této práce je navrhnout a vytvořit takovou aplikaci, která by vycházela ze svých předchůdců, ale poučila se z jejich chyb. Aplikace by měla být jednoduchá na ovládání, dobře čitelná a jasně pochopitelná. Cílem není nabízet všemožné sofistikované funkce, ale spíše umožnit uživateli rychle se v aplikaci zorientovat [9, 12]. Dalším cílem je propagace hry Set. Když si aplikaci stáhne uživatel, který hru nezná, aplikace by mu měla pomoci pochopit princip a pravidla hry, aby mohl začít hrát bez dlouhého vysvětlování a studování.

Následující kapitola podrobně rozebírá pravidla hry a různé situace, které při hře mohou nastat, a studuje podobné existující aplikace. Další kapitola shrnuje důležité možnosti vývoje aplikací pro Android, které budou v práci využity. Kapitola 4 popisuje návrh aplikace, to znamená, jak by měla aplikace vypadat a jak se chovat. Poslední kapitola prochází samotnou realizací práce od vytvoření základních prvků přes rozpracování detailů až po testování a zveřejnění výsledné aplikace.

Kapitola 2

Rozbor hry Set

Hra Set [3] je logická karetní hra, kterou vytvořila Marsha Falco v roce 1974. Hra se dostala do povědomí veřejnosti až po svém zveřejnění v roce 1991. Od té doby se hra rozšířila a získala velký počet příznivců, dočkala se mnoha verzí, různých úprav a adaptací. Také byla předmětem vědeckého bádání [8]. Na základě této hry bylo vytvořeno také několik mobilních aplikací, z nichž je možné vybrat pozitivní přínos a poučit se z jejich nedostatků. Existujícími adaptacemi hry se zabývá druhá část této kapitoly. Před vytvořením nové aplikace je třeba přesně definovat pravidla hry. Tato pravidla, průběh a různé další vlastnosti hry Set zkoumá první část této kapitoly.

2.1 Pravidla hry

Hra Set je karetní hra. Používají se speciální karty vytvořené pro tuto hru. Smyslem hry je hledat trojice karet splňující určitá pravidla. Klasická forma hry se hraje ve skupině hráčů v reálném čase (tzn. bez tahů).

2.1.1 Popis herní karty

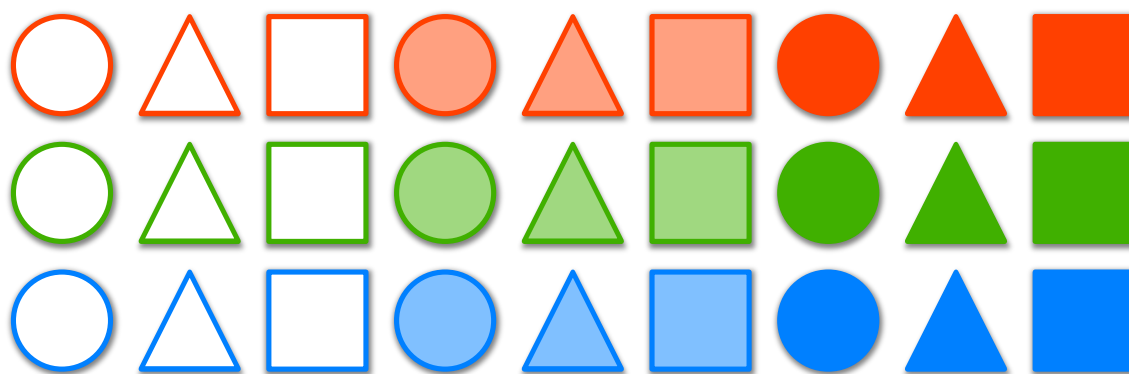
Pro hru jsou vytvořeny speciální karty. Obrazce na lícové straně karet mají čtyři základní vlastnosti. Barvu, počet, výplň a tvar. Každá vlastnost nabývá jedné ze tří možných hodnot.

- Barva: červená, zelená (žlutá¹), modrá (purpurová¹)
- Počet: jeden, dva, tři
- Výplň: prázdná, poloprůhledná (šrafovaná¹), plná
- Tvar: kruh (ovál¹), trojúhelník (vlnovka¹), čtverec (kosočtverec¹)

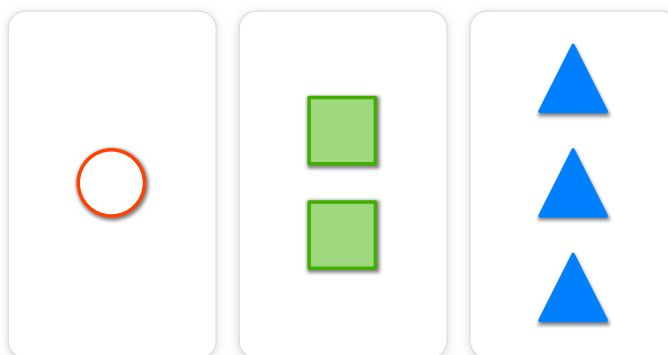
V jedné hře se všechny karty vyskytují právě jednou, tedy v jednom balíčku je 81 karet². Všechny hodnoty různých vlastností (kromě počtu) ukazuje obrázek 2.1. Všechny obrazce na jedné kartě jsou vždy stejné. Příklad karet s různými vlastnostmi ukazuje obrázek 2.2.

¹V různých podáních hry se mohou hodnoty lišit, ale vždy jsou tři pro každou vlastnost.

²Pro 3 hodnoty všech 4 vlastností je $3^4 = 81$ možností.



Obrázek 2.1: Na kartách mohou být různé obrazce (kruhy, trojúhelníky, čtverce), mohou být různě vyplněné (prázdné, poloprůhledné, plné) a mohou mít různou barvu (červenou, zelenou, modrou).



Obrázek 2.2: Příklad karet s různými vlastnostmi.

2.1.2 Podmínky platnosti setu

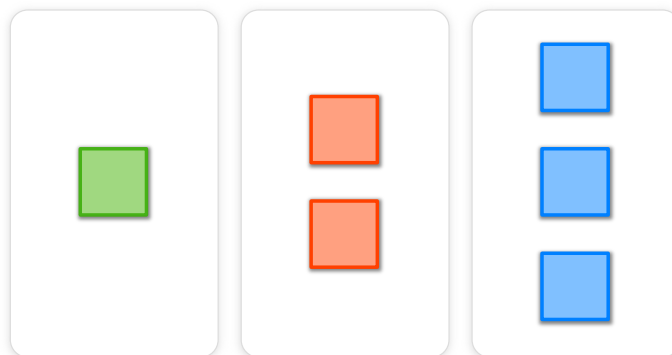
Hra spočívá v hledání trojic karet splňujících určitá pravidla – tzv. setů. Správný set je taková trojice karet, kde je pro každou vlastnost splněna jedna z následujících podmínek, přičemž pro každou vlastnost může být splněna jiná podmínka.

- Všechny tři karty se v dané vlastnosti shodují.
- Žádné dvě karty se v dané vlastnosti neshodují.

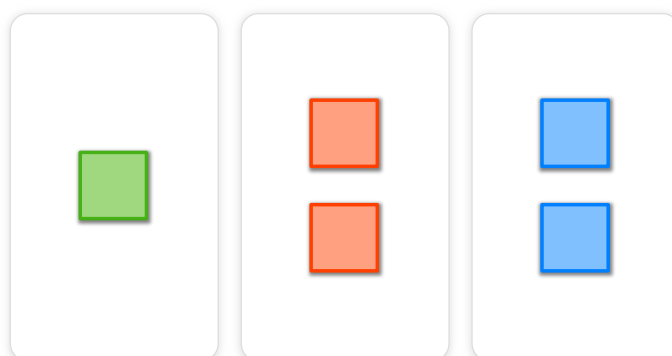
Jinak řečeno, pokud se libovolná vlastnost na dvou kartách shoduje a na třetí je jiná, daná trojice karet není platným setem. Pravidla platného nebo chybného setu ilustrují obrázky 2.3 a 2.4.

2.1.3 Průběh hry

Klasická karetní hra se hraje v libovolném počtu hráčů s papírovými kartami. Na stůl se rozdává devět karet a hráči mezi nimi hledají sety. Jedná se o hru v reálném čase, tedy všichni hráči hledají sety současně. Když některý hráč set najde, ukáže ho ostatním, aby jej schválili, nebo zamítli. Pokud je set schválen, hráč si ho sebere a místo něj se rozdají další tři karty. Hra pokračuje, dokud se nevyčerpá celý balíček karet. Když je balíček vyčerpán, každý hráč si spočítá, kolik setů našel. Vítězí hráč s nejvyšším počtem nalezených setů.



Obrázek 2.3: Příklad platného setu. Tvar a intenzita výplně obrazců je na všech kartách stejná, barva a počet obrazců je na každé kartě jiná.

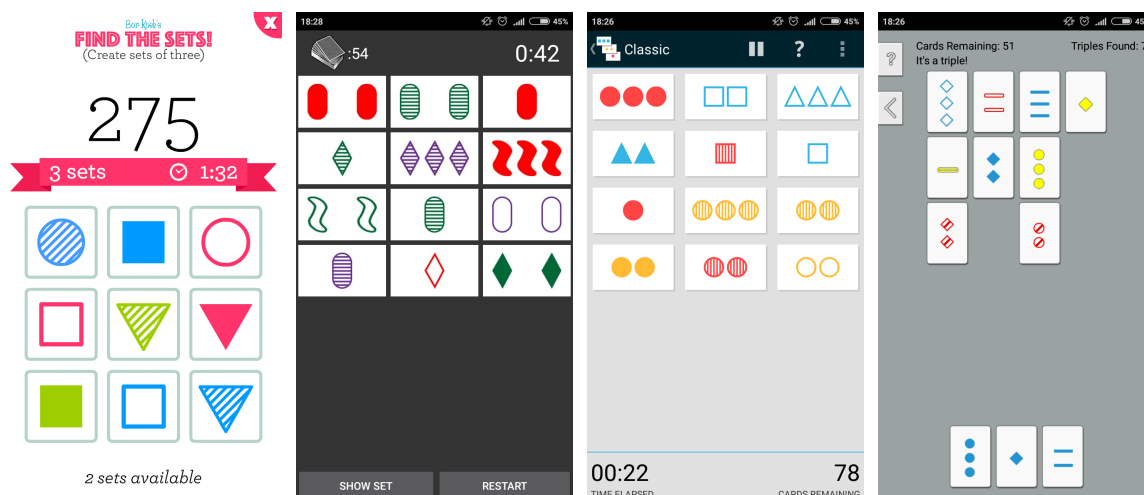


Obrázek 2.4: Příklad chybného setu. Podmínky setu nesplňuje počet obrazců na kartách. Na jedné kartě je jeden obrazec a na ostatních dva. Aby byla tato trojice platným setem, musely by na první kartě být dva čtverce, nebo na jiné kartě tři.

V rozdaných kartách se také nemusí vyskytnout žádný set. Různé výzkumy zabývající se nejvyšším možným počtem karet, mezi kterými není žádný set, shrnuje Erica Klarreich [8]. Podle zmíněných výzkumů se ve skupině do dvaceti karet (včetně) nemusí vyskytnout žádný set. Pokud hráči nemohou mezi rozdanými devíti kartami žádný set najít, jednoduše rozdají další tři. Může se ovšem také stát, že mezi kartami nějaký set je, jen si ho hráči nevšimnou. Vyřešení daného dilematu je jedním z cílů této práce.

2.2 Podobné existující aplikace

Hra Set už byla také několikrát implementována jako aplikace pro Android. Následující kapitoly se věnují některým aplikacím a jejich přínosu pro tuto práci, která je snahou o využití pozitivních stránek předchozích aplikací a odstranění těch méně zdařilých. Snímky obrazovky zkoumaných aplikací ukazuje obrázek 2.5.



Obrázek 2.5: Snímky obrazovky jednotlivých zkoumaných aplikací (Zleva: 1. Find The Sets; 2. Find a set; 3. Tripples; 4. Tripple play)

2.2.1 Find The Sets

Aplikace Find The Sets¹ je zjednodušenou podobou hry Set. Zjednodušení pravidel hry spočívá ve vypuštění jedné vlastnosti (počtu obrazců). Hra má jednoduché rozhraní. Na čtvercových kartách je vždy jeden obrazec a zabírá většinu plochy karty. Obrazce jsou dostatečně velké a dobře viditelné. Negativně by se dalo hodnotit příliš velké číslo udávající počet bodů, které zabírá zhruba pětinu celé obrazovky, nepoměrně oproti dalším důležitým prvkům (např. ukazatel časomíry). Na druhou stranu, i tak je prostředí stále ještě dostatečně přehledné.

Problém s absencí setu mezi rozdanými kartami byl vyřešen vypuštěním vlastnosti počtu obrazců na kartě. Na druhou stranu se tím značně omezí výběr karet. Karty se tedy mohou ve hře opakovat, aby hra neskončila po několika setech. Příjemným prvkem hry je ukazatel, kolik setů se zrovna v rozdaných kartách vyskytuje.

Zajímavou variantou hry na čas je koncept, ve kterém se čas odečítá a hráč musí stihnout posbírat co nejvíce setů. Úspěšnost se pak neměří dosaženým časem, ale počtem setů, které hráč stihl za určený čas nasbírat. Nejlepší dosažené výsledky je možné porovnat globálně se všemi uživateli, toho je však dosaženo za cenu závislosti zobrazení výsledků na připojení k internetu.

Hra poskytuje vysvětlení pravidel doplněné několika různými příklady správných i chybných setů. Příklady jsou pouze spoře vysvětlené. Jako nedostatek hodnotím absenci jakékoli formy tutoriálu, kde by si hráč mohl hru vyzkoušet s možností nápovědy či vysvětlení. Dokonce ani ve hře na čas není žádná možnost nápovědy. Každý hráč je tak odkázán na pochopení pravidel z vysvětlení a několika příkladů.

2.2.2 Find a set

Aplikace Find a set² má velmi jednoduchý, až strohý design. Tlačítka i karty jsou prosté obdélníky a pozadí má jednolitou barvu. Výběr karty je znázorněn jejím silným ohraničením. Nikde nepřekáží nadbytečná okrasná grafika. Karty jsou dobře čitelné, ale mohly by být

¹<https://play.google.com/store/apps/details?id=nl.borkoek.set>

²<https://play.google.com/store/apps/details?id=com.nelis.set>

i větší. Pokaždé je rozdáno minimálně 12 karet. Aplikace používá původní design karet z roku 1974 (ovál, vlnovku a kosočtverec).

Problém s chybějícím setem je vyřešen přidáváním dalších trojic karet a opětovnou kontrolou, zda se set mezi kartami vyskytne. Problém je, že karty musí být menší, aby se vešly na obrazovku i ve větším počtu, a proto je po většinu hry velká část plochy nevyužitá. Další nevýhodou tohoto přístupu je, že pokud se objeví nový řádek, hráč může s jistotou vědět, že minimálně jedna karta hledaného setu se vyskytuje právě v poslední řadě karet. Navíc najít jeden set je tím těžší, čím více karet je rozdáno.

Hra se hraje vždy s měřením času. Hráč si může vybrat ze dvou variant. V první variantě hra končí vyčerpáním celého balíčku karet a v té druhé je hra omezena na prvních 10 setů.

Zásadním nedostatkem aplikace je naprostá absence jakéhokoli vysvětlení pravidel, výukové hry, příkladů a dokonce i nápovědy. Hráč tedy musí hru dobře znát už odjinud, jinak nedokáže aplikaci použít.

2.2.3 Tripples

Grafické prostředí aplikace Tripples³ je na první pohled trochu matoucí. Při spuštění se místo očekávané vstupní nabídky objeví velká prázdná plocha uprostřed. Uživatel musí zjistit, že novou hru spustí malým tlačítkem „+“ u horního okraje obrazovky. Design karet je jednoduchý, využívá (oproti původnímu designu) základní tvary (kruh, trojúhelník a čtverec) a příjemné světlé barvy. Animace simulující pohyb při sbírání karet uhodnutého setu sice vypadají strojově a svým trváním nepatrně prodlužují herní čas, ale zlepšují celkový dojem ze hry přiblížením se realitě. Karty jsou dobře čitelné, ale ovladatelnost hry by mohla zlepšit větší velikost karet pro snazší trefování při rychlém klikání na sety. Navíc je na obrazovce stále spousta volného místa.

Problém s absencí setu ani jeho ošetření nebylo zjištěno. Buďto je existence setu kontrolována programově při rozdávání, nebo v případě absence setu může být využita velká prázdná plocha pro doplnění karet. Při výzkumu se nikdy nestalo, že by mezi dvanácti rozdanými kartami set chyběl.

Zajímavým přínosem této aplikace je, že ukládá každou rozehranou (a nedokončenou) hru. V případě přílišného množství rozehraných her je také uživatel může mazat. Další pozitivní vlastností aplikace je, že hru je možné pozastavit (přestane se přičítat čas a karty jsou skryty). Uživatel má na výběr ze dvou herních módů. V klasickém módu se počítá čas a hra končí vyčerpáním balíčku. V módu arkády má hráč ve stanoveném čase (1 minuta) najít co nejvíce setů.

Pravidla si může nový uživatel přečíst po kliknutí na tlačítko nápovědy. U pravidel je také náhodně generovaný příklad s vysvětlením. Ten je možné obnovit, takže hráč může dostat neomezené množství příkladů, avšak v příkladech se objevují pouze platné sety. Chybné sety na žádném příkladu vysvětleny nejsou. V aplikaci chybí učební mód, kde by hráč mohl sám zkoušet hádat trojice a získal k nim zpětnou vazbu (zda je trojice správně či špatně a proč).

2.2.4 Tripple play

Aplikace Tripple play⁴ má přehledné uživatelské rozhraní s malým počtem nezbytných ovládacích prvků, ale grafika karet už tak přehledná není. Šrafované obrazce jsou spíše

³<https://play.google.com/store/apps/details?id=com.antsapps.triples>

⁴<https://play.google.com/store/apps/details?id=com.iat11.tripleplay>

proškrtnuté jedinou tlustou čarou uprostřed a nejhorší vzhled mají obdélníky, které jsou dlouhé a úzké, takže se na nich těžko poznává, zda jsou vyplněné, prázdné nebo proškrtnuté. Karty se rozdávají a sbírají animovaně.

Na začátku se rozdá 12 karet. Karty se nedoplňují po každém uhodnutém setu, ale pouze tehdy, když už mezi rozdanými kartami žádný set není. Tím je vyřešen problém chybějícího setu, za cenu případného volného místa nebo velkého počtu současně rozdaných karet. Další nevýhodou tohoto přístupu je, že pokud je odebrána trojice karet a další karty nejsou doplněny, rozdané karty se mohou jevit neuspořádaně. Nicméně tento (pro mobilní aplikaci možná nevhodný) způsob rozdávání se nejvíce podobá reálné karetní hře.

Aplikace má dva herní módy. Jeden klasický pro sólovou hru a jeden pro více hráčů. Mód pro více hráčů se odehrává na jednom zařízení. Po každém uhodnutém setu se ještě musí vybrat číslo hráče, který set získal. Jediný rozdíl oproti klasickému módu je tedy oddělené počítání setů pro jednotlivé hráče.

V každém módu hry lze zobrazit stručný přehled pravidel. Navíc aplikace disponuje tréninkovým módem, který funguje tím způsobem, že hráč dostává náhodné trojice karet a pro každou rozhoduje, zda trojice je nebo není platným setem. Pokaždé, když hráč udělá chybu, je mu podáno vysvětlení.

Kapitola 3

Vývoj aplikací pro Android

Nejlepším nástrojem pro tvorbu nativních aplikací pro Android je prostředí Android Studio, které pochází od stejných tvůrců jako samotný systém Android. V následujících kapitolách jsou probrány některé aspekty Android Studia [7] a vývoje aplikací pro Android [10, 11], které mají nějakou souvislost s touto prací a byly použity nebo zvažovány při vývoji cílové aplikace.

3.1 Android Studio

Android Studio je komplexní vývojové prostředí vytvořené vývojáři systému Android pro tvorbu aplikací pro tento operační systém. Android Studio je zdarma k dostání na webu Android Developers¹, Zmíněný vývojářský web obsahuje také mnoho návodů a podrobnou dokumentaci. Většina informací o Android Studiu a možnostech vývoje byla získána právě z tohoto webu.

Vývoj aplikací probíhá ve dvou hlavních rovinách. Funkcionalita aplikací se programuje v jazyce Java. Vzhled uživatelského prostředí a případně různá pomocná data se definují v jazyce XML. Android Studio dále nabízí vlastní grafický editor pro tvorbu a úpravu uživatelského prostředí aplikací. Samozřejmostí ve vybavení Android Studia je také zvýrazňování syntaxe, našeptávač nebo kontrola správnosti zdrojového kódu v reálném čase.

Android Studio poskytuje také dobré prostředí pro ladění aplikací. Umožňuje spravovat virtuální zařízení nebo se napojit na zařízení fyzická. Při ladění aplikace pomůže možnost nastavení breakpointů, krokování kritických částí kódu nebo sledování hodnot proměnných a objektů.

3.1.1 Import vlastní grafiky do Android Studia

Do Android Studia je možné importovat vlastní grafické soubory. Rastrové obrázky zůstanou ve svém původním formátu a mohou být předány nějaké komponentě k tomu určené (např. `ImageView`) nebo vykresleny při programovém provedení vlastní grafiky v metodě `onDraw`. Vektorové obrázky lze také importovat. Android Studio převádí obecné vektorové formáty (např. SVG) do vlastní vektorové reprezentace v jazyce XML. Některé složitější grafické prvky (např. filtry) způsobují při převodu problémy.

¹<https://developer.android.com/studio/>

3.1.2 Uložení a jazyková lokalizace řetězců

Vývojáři Androidu doporučují všechny řetězce použité nejen v uživatelském rozhraní ukládat ve speciálním, k tomu určeném souboru XML [4]. Na místě použití daného řetězce se pak nepíše přímo ten řetězec, ale už jen odkaz do souboru se všemi řetězci. Tento přístup má několik výhod. Jednak mnoho řetězců se vyskytuje v jednom programu vícekrát a při jejich změně hrozí opomenutí některého z výskytů měněného řetězce, aplikace může potom vypadat (nebo se chovat) nekonzistentně.

Další výhodou je možnost definice více jazykových mutací vybraných textů. Jeden jazyk je vždy výchozí a použije se pro všechna zařízení, do jejichž jazyka není v aplikaci vytvořený překlad. Řetězce v různých jazycích se ukládají do různých souborů, takže přidání nového překladu je snadno proveditelné přidáním nového souboru. V aplikaci se také mohou vyskytnout řetězce, které se nemusí nebo nesmí překládat (např. klíče hodnot v souborech Shared Preferences). Takovým řetězcům stačí vypnout příznak přeložitelnosti nastavením atributu `translatable="false"` v souboru XML, kde jsou definovány.

Řetězce je možné ukládat také v podobě formátovacích řetězců a potřebné údaje, které jsou známy až v době běhu programu, se do řetězců doplní při jejich načítání. Do řetězců je takto možné zařadit např. různá dynamická čísla.

3.2 Tvorba grafického uživatelského rozhraní

Při tvorbě aplikací a obzvláště her je třeba klást velký důraz na tvorbu uživatelského rozhraní. Tato kapitola se věnuje některým nástrojům, prvkům a způsobům tvorby uživatelského rozhraní, kterými jsem se zabýval.

3.2.1 Tvorba vlastních grafických komponent

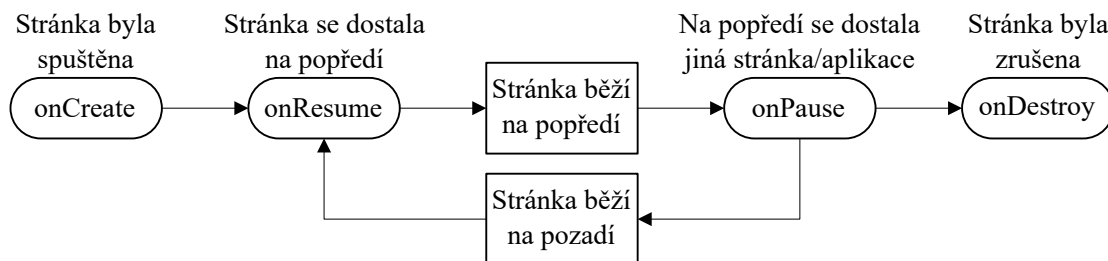
Základní komponentou grafického uživatelského prostředí je komponenta **View**. Vlastní nové komponenty [1] lze vytvářet s pomocí dědičnosti rozšířením třídy **View**. Většina třídy **View** zůstane zachována, což poskytuje rozhraní pro systém Android. Přepsáním některých zděděných metod (`onDraw`, `onMeasure`, `onKeyDown`, `onTouchEvent`) je možné definovat vzhled komponenty a její reakce na akce uživatele. Dále lze třídě přidat všechny potřebné vlastnosti.

Uživatelské rozhraní se v Android Studiu definuje v jazyce XML nebo v grafickém editoru. Vlastní komponenty je možné ovládat úplně stejně jako ty původní. Aby bylo možné nastavovat vlastnosti vlastní komponenty, musí být nadefinovány jako její atributy ve speciálním souboru XML. V tom se nastavuje jméno a typ (přípustné hodnoty) požadovaných atributů XML.

3.2.2 Animace a přechody grafických prvků

Vývojářské nástroje pro Android nabízí mnoho způsobů, jak animovat grafické prvky uživatelského prostředí². Jedním z nich je metoda `View.animate`, která vrací objekt typu `ViewPropertyAnimator`. Ten umožňuje nastavit, která vlastnost grafické komponenty bude animována. To může být některá souřadnice, hodnota průhlednosti, různé transformace a mnohé další. Je možné animovat i více vlastností současně. `ViewPropertyAnimator` také

²<https://developer.android.com/training/animation/>



Obrázek 3.1: Zjednodušené schéma životního cyklu jedné stránky aplikace. Obdélníky znázorňují stavy stránky, šipky přechody mezi stavy a v oválech jsou uvedeny metody volané při přechodu mezi stavy.

umožňuje nastavit parametry animace, jako je čas trvání animace, počáteční zpoždění nebo třeba akce vykonaná po skončení animace.

Další způsob animování se dá nastavit v XML. Je možné nadefinovat několik vektorových objektů (drawables) a ty pak spojit do jednoho přechodu, v němž se hodnoty vektorů postupně mění. Takový nadefinovaný přechod je pak možné programově spustit a případně i upravit.

3.2.3 Podpora pro stínování material designu

Neodmyslitelnou součástí material designu³ je stínování objektů, které znázorňuje vyvýšení plochého objektu nad podložkou. Zdánlivá výška objektu není jen designovým prvkem, ale obvykle také nese určitý význam. Stínování lze nastavovat⁴ pomocí vlastností `elevation` a `translationZ`. Obě tyto hodnoty znamenají míru vyvýšení a výsledné vyvýšení je jejich součtem. Hodnoty vyvýšení je také možno animovat. Podpora material designu (a tedy i stínování) funguje v SDK od úrovně API 21 (viz tabulka 3.1 v kapitole 3.3.3).

3.3 Další aspekty vývoje aplikací

Tato kapitola sdružuje několik dalších aspektů vývoje aplikací pro Android, se kterými jsem se během vývoje setkal. Jedná se o kompatibilitu aplikace s různými verzemi systému, správu různých událostí v rámci jednotlivých stránek aplikace, ukládání dat a publikování hotových aplikací.

3.3.1 Životní cyklus jedné stránky aplikace

Každá aplikace se skládá z jedné nebo více stránek. Tyto stránky se v prostředí Androidu nazývají `Activity` a jejich chování se definuje ve stejnojmenné třídě. Každá stránka má vlastní životní cyklus [5] skládající se z několika stavů. V přechodech mezi stavy jsou volány některé předdefinované metody (`onCreate`, `onResume`, `onPause`, `onDestroy` a další), jejichž přepsáním se dá odchytnout událost přechodu do jiného stavu a zareagovat tak na změnu. Životní cyklus jedné stránky zjednodušeně ilustruje obrázek 3.1.

³<https://material.io/>

⁴<https://developer.android.com/training/material/shadows-clipping>

3.3.2 Ukládání krátkých perzistentních dat

Někdy je potřeba uložit krátká data, která přetrvají i po vypnutí aplikace. K tomuto účelu je možné použít malé soubory označované jako Shared Preferences [2]. Data jsou v těchto speciálních souborech uložena v párech jako klíč a hodnota. Klíčem je vždy řetězec a hodnota může mít několik různých datových typů (`boolean`, `float`, `int`, `long`, `String`, `Set<String>`). K manipulaci s nimi slouží třída `SharedPreferences`, která umožňuje čtení dat, zápis nebo přepis hodnot i mazání záznamů.

3.3.3 Verze SDK

Software development kit (dále SDK) je sada jazykových nástrojů použitelných pro vývoj aplikací. Jednotlivé verze SDK (označované jako úroveň API [6]) odpovídají verzím systému Android a každá verze přináší nová vylepšení. Programátor musí hledat kompromis mezi použitím lepších a novějších možností a dostupností aplikace pro uživatele se staršími zařízeními. Čím novější funkce aplikace používá, tím vyšší verzi SDK vyžaduje, proto je pak dostupná menšímu počtu uživatelů. Používanost jednotlivých verzí ukazuje tabulka 3.1. Údaje v tabulce poskytuje Android Studio při vytvoření nového projektu.

Tabulka 3.1: Jednotlivé verze SDK (úrovně API) odpovídají verzím systému Android. Sloupceček Uživatelé udává přibližný počet uživatelů, kteří mají danou verzi nebo novější. Tedy počet uživatelů obslužitelných danou verzí SDK. Hodnota je založena na počtu uživatelů aktivních na Google Play.

| Android | Označení | Úroveň API | Uživatelé |
|---------|--------------------|------------|-----------|
| 4.0 | IceScream Sandwich | 14 | 100,0 % |
| 4.0.3 | IceScream Sandwich | 15 | 100,0 % |
| 4.1 | Jelly Bean | 16 | 99,2 % |
| 4.2 | Jelly Bean | 17 | 96,0 % |
| 4.3 | Jelly Bean | 18 | 91,4 % |
| 4.4 | KitKat | 19 | 90,1 % |
| 4.4W | KitKat Wear | 20 | 71,3 % |
| 5.0 | Lollipop | 21 | 71,3 % |
| 5.1 | Lollipop | 22 | 62,6 % |
| 6.0 | Marshmallow | 23 | 39,3 % |
| 7.0 | Nougat | 24 | 8,1 % |
| 7.1 | Nougat | 25 | 1,5 % |
| 8.0 | Oreo | 26 | 1,0 % |

3.3.4 Publikování aplikace na Google Play

Aplikace pro Android je možné dostat k uživateli několika cestami. Ta oficiální je přes obchod Google Play⁵. Aby mohl člověk přidávat aplikace na Google Play, potřebuje vývojářský účet. Zařízení takového účtu lze provést přímo na webu Google Play za cenu jednorázového poplatku 25 amerických dolarů.

⁵<https://play.google.com/>

Pro vkládání a správu aplikací na Google Play slouží webová aplikace Google Play Console⁶. Z vytvořené aplikace je třeba pomocí Android Studio vygenerovat soubor APK, který slouží jako instalační soubor aplikací pro Android. Na Google Play Console se pak nahraje instalační soubor APK a přidá se logo, název a popis aplikace, snímky obrazovky aplikace, případně propagační video nebo další materiály. Google Play Console slouží také ke správě verzí aplikace a umožňuje zveřejňovat verze ve čtyřech kanálech (kanál pro interní testování, kanál pro alfa-testy, kanál pro beta-testy a kanál pro produkční vydání). Google Play Console poskytuje také spoustu dalších nástrojů, např. nástroje pro sbírání statistik, propagaci a mnohé další.

⁶<https://play.google.com/apps/publish/>

Kapitola 4

Návrh řešení

Podstatná část návrhu aplikace vznikla už v rané fázi vývoje, nicméně v průběhu vývoje prošla aplikace i mnohými podstatnými změnami. Tato kapitola popisuje spíše finální podobu návrhu. Zaměřuje se hlavně na to, jak jednotlivé části aplikace vypadají, co dělají a jak se chovají, než na konkrétní způsob, jakým toho bylo dosaženo. Použité postupy a způsob řešení rozebírá až další kapitola. Tato kapitola se věnuje celkovému konceptu aplikace, návrhu uživatelského rozhraní, logice hry a dalším operacím, které jsou uživateli skrze uživatelské rozhraní zpřístupněny.

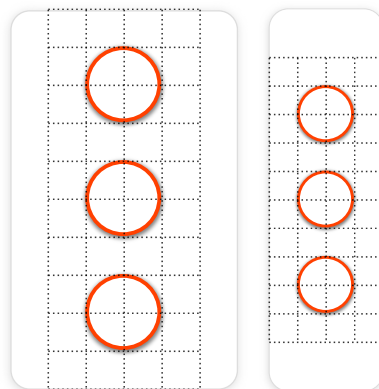
4.1 Uživatelské rozhraní

Hlavní částí uživatelského rozhraní je zobrazení hry. Při hře bude rozdáno pokaždé devět karet, bude se hráči ukazovat ubíhající čas a bude mu umožněno vyžádat si nápovědu. Veškerá grafika aplikace by se měla blížit koncepci material design navržené společností Google. Základní prvky vypadají jako plochy pozvednuté nad povrch (viz kapitola 3.2.3), grafika je jednoduchá a výrazná a změny v grafickém prostředí nejsou tvrdé skoky, ale jsou měkce animované pro zachování plynulosti. Aplikace bude obsahovat také stránku s vysvětlením pravidel, které bude doplněno okomentovanými příklady.

4.1.1 Zobrazení herní karty

Ústředním prvkem hry je herní karta (viz kapitola 2.1.1), proto je třeba jí věnovat náležitou pozornost. Komponenta znázorňující herní kartu by měla připomínat skutečnou fyzickou kartu. Pozadí karty bude bílé a její rohy zakulacené, kartu zvýrazní mírné vyvýšení znázorněné pomocí stínu.

Na kartě jsou zobrazeny obrazce (kruhy, trojúhelníky, čtverce) a může jich být různý počet. Obrazce by měly být výrazné a dobře viditelné, tedy dostatečně velké, ale nevypadalo by pěkně kdyby byly namačkané na sobě. Stanovil jsem, že volná místa mezi obrazci nebo kolem nich budou mít oproti samotným obrazcům poloviční velikost. Všechny obrazce by měly být stejně velké, ať už budou na kartě tři, nebo jen jeden. Jejich velikost se počítá tak, aby tři obrazce společně se stanoveným místem mezi nimi a kolem nich zabraly celou plochu karty. Pokud je obrazců na kartě méně, mají stejnou velikost i stejné případné rozestupy, jen jsou zarovnány na střed karty. Výpočet velikostí obrazců a jejich rozmístění znázorňuje obrázek 4.1.



Obrázek 4.1: Velikost a rozložení objektů na kartě závisí na rozměrech karty. Základní jednotkou při výpočtu rozměrů a rozložení je $\frac{1}{4}$ šířky nebo $\frac{1}{10}$ výšky. Pokud se nerovnájí, vybere se menší z nich.

4.1.2 Uspořádání grafických prvků hry

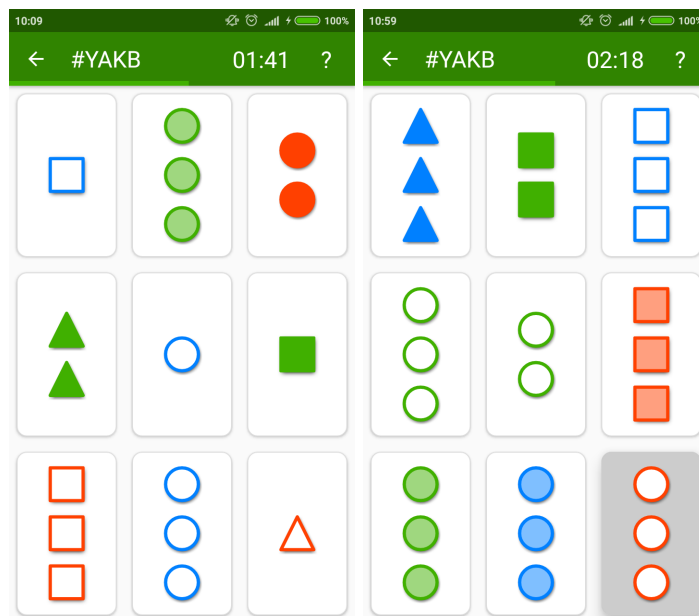
Na herní plochu je rozdáno vždy 9 karet a jsou poskládány v mřížce 3×3 karty. Na rozdíl od většiny podobných již existujících aplikací (viz kapitola 2.2) je dbáno na maximální využití plochy obrazovky, aby byly karty dostatečně velké a co nejlépe viditelné. Větší velikostí karet se také usnadní trefování kliknutí na danou kartu při vyšší rychlosti klikání v zápalu hry. Karty tedy zabírají většinu plochy displeje a dalších několik nezbytných prvků je soustředěno v horní liště aplikace.

V horní liště se nachází šipka zpět, kód hry, ukazatel času a ukazatel postupu hry. Šipka zpět umožňuje návrat do hlavní nabídky. Návrat je možný i pomocí systémových nebo hardwarových kláves zařízení. Kód hry udává konkrétní zamíchání karet. Asi nejdůležitější je ukazatel herního času. Ten také ukazuje časové penalizace za chybný set. Postup hry, tedy jak velká část setů už byla vybrána, zobrazuje zvětšující se světle zelená čára na spodním okraji aplikační lišty. Celé rozložení grafických prvků aplikace ukazují snímky herní obrazovky na obrázku 4.2.

4.1.3 Dynamika grafických prvků hry

Karty se vybírají kliknutím (dotykem) na ně. Vybraná karta se šedě podbarví a zvětší se její vyvýšení symbolizované stínem (viz kapitola 3.2.3). Pozadí i vyvýšení se mění plynule s použitím animace/přechodu (viz kapitola 3.2.2). Když jsou vybrány tři karty, aplikace vyhodnotí, zda se jedná o platný set. Pokud byl vybrán platný set, karty zmizí a na jejich místě se objeví další. Ke zmizení a objevení karet jsou také použity animace. Karty odletí do levého dolního rohu (jakoby směrem k hráči) a z protějšího – pravého horního rohu přiletí nové karty. Animace karet by měla imitovat pohyb reálných fyzických karet.

Pokud je zvolená trojice chybná (nejedná se o platný set), všechny tři karty naráz zčervenají a poté se postupně vrátí do původního stavu s užitím barevného přechodu. Zároveň také ukazatel času překryje ukazatel penalizace, který oznamuje, že hráč dostal 10 trestných sekund. Stejně jako je pro chybný set použit barevný přechod z červené zpět do původní bílé, podobný přechod používá i nápověda. Po kliknutí na tlačítko nápovědy se zvýrazní karty navrhovaného setu zeleným pozadím, které se opět postupně vrátí k původní bílé barvě.



Obrázek 4.2: Většinu herní obrazovky zabírá 9 karet v mřížce 3×3 karty. Další prvky uživatelského rozhraní hry jsou soustředěné v horní liště. Výběr karty je znázorněn větším vyvýšením (stínem) a šedým podbarvením karty.

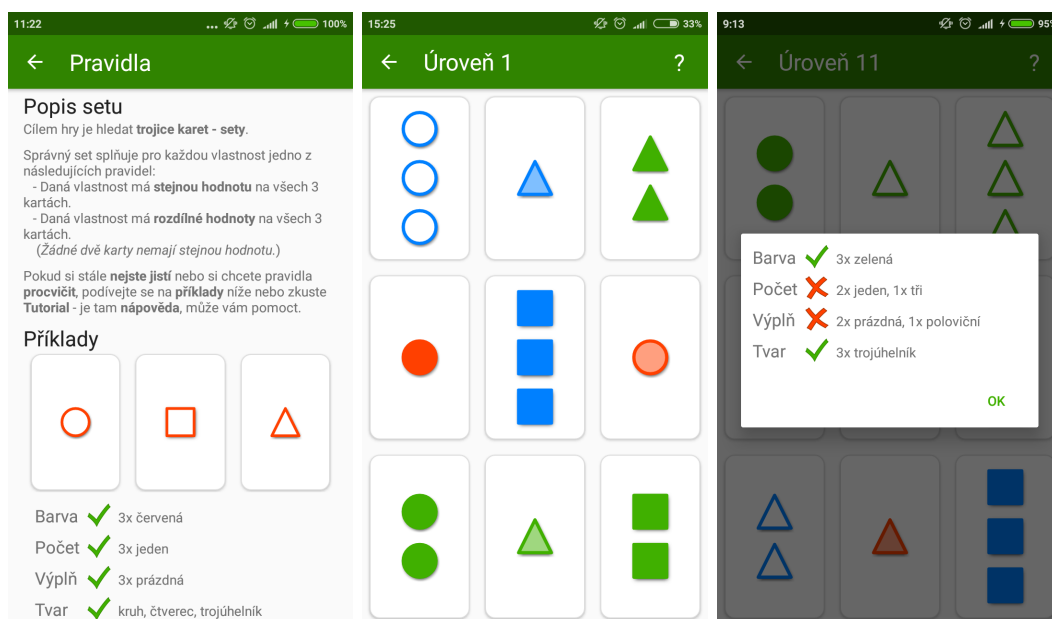
4.1.4 Vysvětlení pravidel, příklady a tutoriál

Aplikace má poskytovat začínajícím hráčům možnost naučit se pravidla hry. Dále by jim měla pomoci hraní hry jednoduše procvičit. Pokud uživatel hned na začátku narazí na těžký problém, aplikaci vypne a už se k ní nevrátí. První možností pochopení pravidel, kterou aplikace nabízí, je shrnutí pravidel, které obsahuje stručný popis herní karty a podmínek setu. Shrnutí pravidel je předáno v podobě formátovaného textu. Pravidla jsou doplněna několika vybranými příklady platných i chybných setů. Ke každému příkladu je dodáno vysvětlení, proč je zrovna tato trojice platným nebo chybným setem. Vysvětlení graficky i slovně ukazuje, proč některé vlastnosti splňují nebo nesplňují podmínky setu.

K procvičení pravidel hry pak slouží tutoriál, který funguje stejně jako klasická hra až na pár drobných, leč podstatných odlišností. V tutoriálu se nepočítá čas a při zvolení chybného setu je hráči podáno vysvětlení, které vlastnosti porušují podmínky setu a jakým způsobem. Hráč má možnost vyžádat si nápovědu, která má v tutoriálu tři stupně:

1. Zeleně problikne jedna ze tří karet navrhovaného setu.
2. Hráči se zobrazí vysvětlení, které vlastnosti navrhovaného setu jsou shodné a které se navzájem liší. (Vysvětlení v nápovědě vypadá podobně jako vysvětlení chyby.)
3. Zeleně probliknou všechny tři karty navrhovaného setu.

Tutoriál je rozdělen do 15 úrovní podle náročnosti. V každé úrovni se při rozdávání nebo doplňování karet volí cíleně právě takové sety, jejichž náročnost odpovídá zvolené úrovni. Obrázek 4.3 ukazuje snímek herní obrazovky tutoriálu a vysvětlení chyby.



Obrázek 4.3: Aplikace poskytuje stránku s vysvětlením pravidel a několika příklady. K procvičení může hráč využít tutoriál. Ten se od klasické hry liší jen v drobných detailech. Neměří se čas, vybírají se sety podle zvolené úrovně náročnosti a při chybné volbě setu se hráči zobrazí vysvětlení jeho chyby.

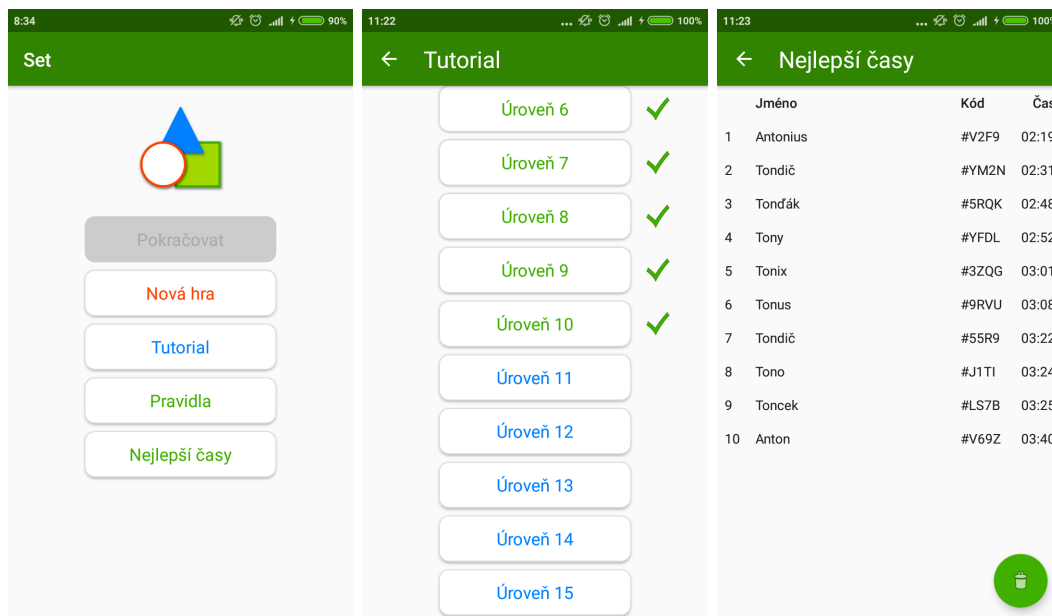
4.1.5 Ostatní části aplikace

První stránka, která se zobrazí po spuštění aplikace, je úvodní nabídka. Pro konzistentnost vzhledu celé aplikace mají tlačítka vstupní nabídky podobný vzhled jako karty. Mají kulaté rohy, jsou mírně vyvýšené nad povrch a text tlačítek má stejné barvy jako barvy obrazců na kartách. Stejně jako karty, tlačítka reagují na kliknutí (dotyk) ztmavnutím a vyvýšením. Jediné, v čem se tlačítka od karet liší je poměr jejich stran. Karty jsou postaveny na výšku a tlačítka v nabídce na šířku.

Vstupní nabídka obsahuje tlačítka pro založení nové hry, pokračování v rozehrané hře, spuštění tutoriálu, zobrazení pravidel hry a nejlepších dosažených výsledků. Barvy textu tlačítek jsou odlišeny podle jejich účelu. Tlačítka týkající se hry na čas mají text červený, tlačítko pro tutoriál modrý a další tlačítka vedoucí na informační stránky mají text zelený. Tlačítko pro pokračování v rozehrané hře je neaktivní, pokud právě žádná hra není uložena. V takovém případě je celé tlačítko šedé včetně textu.

Po kliknutí na tlačítko tutoriálu se zobrazí nová nabídka, která hráči umožňuje vybrat si úroveň obtížnosti a také ukazuje, které úrovně už hráč prošel celé až do konce. Tlačítka úrovní mají stejný vzhled jako tlačítka úvodní nabídky a jejich text je modrý. Tlačítka odkazující na již splněné úrovně mají text zelený a vedle nich se objevuje zelená značka „splněno“.

Aplikace umožňuje zobrazit 10 nejlepších dosažených výsledků. Ty se zobrazují v tabulce, která v jednotlivých sloupcích udává pořadí daného výsledku, jméno hráče, který výsledku dosáhl, čtyřmístný kód hry a dosažený čas. Na stránce s nejlepšími časy se také nachází tlačítko umožňující všechny záznamy smazat. Editovat záznamy jednotlivě není možné.



Obrázek 4.4: Různé stránky aplikace, dále uvedeno zleva: Jednotlivá tlačítka vstupní nabídky mají vzhled podobný herním kartám. Nabídka úrovní obtížnosti tutoriálu ukazuje, které úrovně už hráč prošel celé. Zobrazení nejlepších dosažených výsledků ukazuje nejvýše deset záznamů v tabulce a umožňuje záznamy smazat (tlačítko vpravo dole).

4.2 Herní logika

Herní logika je rozdělena do třech tříd. První třída **Card** symbolizuje jednu kartu a uchovává její vlastnosti. Vlastnosti jsou jednou inicializované a poté už je nelze měnit. Třída **Card** poskytuje také několik statických metod pro ověření nebo nalezení setu. Další třída **SetGame** znázorňuje jednu hru, uchovává údaje o rozdaných kartách i kartách v balíčku a provádí všechny potřebné herní operace, jako je např. míchání nebo rozdávání karet. Poslední třída **TutorialGame** upravuje a rozšiřuje třídu **SetGame** pro potřeby tutoriálu.

4.2.1 Logika karty

Logiku karty definuje třída **Card**, která uchovává vlastnosti karty (barva, počet, výplň a tvar). Jediný konstruktor třídy **Card** požaduje zadání všech čtyř vlastností jako vstupních parametrů. Třída poskytuje přístup ke svým vlastnostem, ale pouze pro čtení, vlastnosti po vytvoření nové karty nelze měnit.

Třída **Card** nabízí několik statických metod operujících s kartami. První a nejdůležitější z nich je metoda **isSet** pro ověření platnosti setu. Vstupem této metody jsou tři karty a výstupem je logická hodnota udávající, zda jsou dané tři karty planým setem. Na této první metodě je postavena další metoda **findSet**, která očekává libovolné pole karet jako vstupní parametr a mezi danými kartami najde jeden set. Výstupem je pole tří čísel, která jako indexy ukazují do původního vstupního pole na nalezený set. V případě, že žádný set není nalezen, je vrácena prázdná hodnota **null**.

Další statická metoda **toASCII** očekává jednu kartu jako vstupní parametr a umožňuje převést kartu na ASCII znak pro potřeby ukládání karet do souborů. Máme 81 možných kombinací vlastností karet a základní tisknutelné ASCII znaky začínají hodnotou 32 a končí

u hodnoty 126. Rozsah karet se do rozsahu ASCII znaků bez problému vejde a tento způsob ukládání je velmi úsporný. Metoda vrací hodnotu typu `char`. Poslední metoda `fromASCII` funguje opačně než metoda `predchozi` a převede znak zpět na kartu.

4.2.2 Logika jedné hry

Hlavní třída, která definuje logiku hry je třída `SetGame`. Tato třída v sobě nese údaje o stavu hry. Jsou zde uloženy karty v balíčku i ty rozdané. Třída také měří čas hry a provádí téměř všechny herní operace. Objekty třídy `SetGame` je možné vytvořit jedním ze tří konstruktorů. První konstruktor vygeneruje náhodnou hru, naplní a zamíchá kompletní balíček karet a prvních devět rozdává. Další konstruktor umožňuje načíst dříve rozehranou a poté uloženou hru podle řetězce načteného ze souboru. Karty jsou přečteny z řetězce a zařazeny tak, jak byly dříve uloženy. Poslední konstruktor umožňuje začít novou hru podle zadaného kódu hry. Kód hry je čtyřmístný řetězec znaků 0–9 nebo A–Z a udává konkrétní permutaci karet v balíčku po počátečním zamíchání.

Nejdůležitější metodou třídy `SetGame` je metoda `guessSet`. Tato metoda přijímá jako vstupní parametry tři celá čísla, která indexují rozdané karty. Dále kontroluje, jestli daná trojice karet tvoří platný set. Výsledkem metody je logická hodnota označující, zda je set platný nebo ne, ale metoda toho obsluhuje mnohem víc. V případě neplatného setu zajišťuje časovou penalizaci za chybu a při platném setu se musí postarat o doplnění odebraných karet. Při rozdávání nebo doplňování karet je nutné zajistit, aby mezi rozdanými kartami byl vždy nějaký set. Pokud už není možné rozdat karty tak, aby mezi nimi byl set, hra končí.

Metoda `getBoard` zpřístupňuje rozdané karty, aby bylo možné je v uživatelském prostředí zobrazit. Karty v balíčku není třeba zpřístupňovat, stačí znát velikost zbývajících balíčku, k tomu slouží metoda `getStockSize`. Čas uplynulý od začátku hry pro zobrazení časomíry vrací metoda `getTimeElapsedMillisec`. Metoda `isGameOver` vrací logickou hodnotu, která v kladném případě signalizuje konec hry. Tu je potřeba zkontrolovat po každém uhodnutém setu. Čtyřmístný kód hry vrací metoda `getHash`, která se obvykle volá pouze jednou na začátku hry, neboť kód se v průběhu hry nemění. Poslední důležitou metodou je metoda `getStateString`, která vrací jeden řetězec o délce do sta znaků, ve kterém je obsažen kompletní stav hry, tedy rozdané karty i karty v balíčku. Tato metoda se používá při ukládání hry.

4.2.3 Logika tutoriálu

Logika výukové hry (tutoriálu) je do značné míry podobná logice klasické hry. Logiku tutoriálu definuje třída `TutorialGame`, která je potomkem třídy `SetGame` a rozšiřuje ji o některé funkce a vlastnosti, případně některé existující upravuje. Třída nabízí dva konstruktory. První klasický vyžaduje předání jednoho parametru, kterým je celé číslo označující úroveň obtížnosti. Druhý konstruktor umožňuje navázat na některou dříve rozehranou výukovou hru a kromě úrovně obtížnosti vyžaduje také textový řetězec s uloženými kartami podobně jako konstruktor pro načítání klasické hry. Co se týče logiky, čas se v tutoriálu počítá stejně jako v klasické hře, ale není k ničemu použit. Počítání času tu bylo ponecháno pouze proto, že by bylo náročné a navíc úplně zbytečné ho uměle rušit.

Tutoriál je rozdělený do úrovní podle obtížnosti. Úrovní je dohromady 15 a jsou určeny přibližnou náročností různých setů. Náročnost setu je samozřejmě velmi subjektivní záležitost, která závisí nejen na vnímání každého člověka, ale také na rozmístění karet, na ostatních kartách kolem i na předchozích uhodnutých setech. Přibližná náročnost setu vy-

Tabulka 4.1: Úrovně obtížnosti setů podle vztahů mezi vlastnostmi karet. Rozdělení do úrovní vychází z předpokladu, že sety, u kterých se více vlastností na všech kartách shoduje jsou snáze odhalitelné než ty, u kterých se více vlastností na každé kartě liší.

= Daná vlastnost je stejná na všech kartách.

≠ Daná vlastnost je na každé kartě jiná.

| Úr. | Vlastnosti | | | |
|-----|------------|-------|-------|------|
| | Barva | Počet | Výplň | Tvar |
| 1 | = | = | = | ≠ |
| 2 | = | = | ≠ | = |
| 3 | = | ≠ | = | = |
| 4 | ≠ | = | = | = |
| 5 | = | = | ≠ | ≠ |
| 6 | = | ≠ | = | ≠ |
| 7 | ≠ | = | = | ≠ |
| 8 | = | ≠ | ≠ | = |
| 9 | ≠ | = | ≠ | = |
| 10 | ≠ | ≠ | = | = |
| 11 | = | ≠ | ≠ | ≠ |
| 12 | ≠ | = | ≠ | ≠ |
| 13 | ≠ | ≠ | = | ≠ |
| 14 | ≠ | ≠ | ≠ | = |
| 15 | ≠ | ≠ | ≠ | ≠ |

chází z předpokladu, že snáze objevitelný set je takový, který má více vlastností shodných. Naopak náročnější sety jsou takové, u kterých je na každé ze tří karet více vlastností různých. Některé vlastnosti jsou vyhodnoceny jako výraznější a některé jako méně výrazné. Kompletní určení úrovní obtížnosti ukazuje tabulka 4.1.

V klasické hře se při rozdávání karet kontroluje výskyt alespoň jednoho setu mezi rozdanými kartami. V tutoriálu je mechanismus rozdávání karet upravený tak, že kontroluje výskyt alespoň jednoho setu dané úrovně odpovídající zvolené úrovni tutoriálu. Stejně tak hledání setu při vyžádání nápovědy musí vrátit právě set s danou úrovní obtížnosti.

4.3 Ukládání dat

Některá data je třeba uchovat i po vypnutí aplikace. Jedním z takových případů je ukládání rozehrané hry. Současná hra se ukládá automaticky, aby v případě přerušení hry bylo možné v ní později pokračovat tam, kde se skončilo. Je třeba ukládat rozdané karty a také karty v balíčku, a to se zachováním původního pořadí. Dále je třeba ukládat uplynulý čas, aby při obnovení hry mohla časomíra navázat na moment, kdy byla hra přerušena. Posledním důležitým údajem, který je třeba uložit, je kód hry. Ten má sice praktický význam pouze na začátku hry, ale je stále potřeba ho znát i při obnovení hry, jelikož kód musí být u hry vždy napsán, aby uživatel měl možnost vědět, jakou hru hraje. Ve vstupní nabídce se pokaždé kontroluje, zda nějaká uložená hra existuje a podle toho je nebo není zpřístupněno tlačítko

pro obnovení uložené hry. Hra je uložena po každém přerušení. Pokud je hra dohrána do konce, stav hry se už neukládá, ale uložená data jsou smazána.

Hry v tutoriálu se ukládají také, jen o nich není třeba ukládat kód hry ani uplynulý čas, jelikož tyto údaje nemají v tutoriálu žádné použití (přestože třída `TutorialGame` tyto hodnoty zdědila). V tutoriálu se ukládá každá úroveň zvlášť a hráč nemá na výběr, jestli chce hrát vybranou úroveň tam, kde skončil, nebo od začátku. To znamená, že se vždy načte uložená hra. Když hráč danou úroveň dokončí, může ji kdykoli začít znovu. V tutoriálu se navíc ukládají údaje o dokončení jednotlivých úrovní. Původním záměrem bylo, že hráč může spustit pouze několik prvních úrovní a k dalším se dostane až po splnění těch prvních, ale nakonec bylo od tohoto záměru upuštěno a všechny úrovně obtížnosti jsou hráči přístupné od začátku.

Další sadou údajů, které je třeba ukládat, jsou nejlepší dosažené výsledky. O nich se ukládají následující informace. Nejdůležitějšími daty jsou jméno hráče a čas, kterého hráč dosáhl. Dále se ukládá i kód hry, aby byla možnost začít hru podle stejného kódu a porovnat se tak s dříve dosaženým dobrým výsledkem. Na jednom zařízení může být uloženo nejvýše 10 her. Dokud je záznamů méně než 10, je možné uložit prakticky každou dohranou hru. Výsledky jsou seřazeny od nejlepšího. Po uložení 10 záznamů se lepší výsledek zařadí mezi uložené a všechny horší výsledky se posunou o jednu příčku níže. Poslední dosavadní záznam je zahozen. Záznamy o nejlepších výsledcích není možné nijak upravovat, ale lze je všechny naráz smazat a žebříček tak resetovat.

Kapitola 5

Implementace

Tato kapitola sleduje vývoj aplikace od jeho začátku přes testování částečně hotové aplikace až po její zveřejnění. První část této kapitoly rozebírá, jaké problémy byly v různých fázích vývoje řešeny, jaká rozhodnutí následně byla učiněna a jaké to mělo důsledky. Další část kapitoly se věnuje testování aplikace. Kromě způsobu testování jsou zde uvedena různá zjištění, která z testů vyplynula a také případné reakce na daná zjištění. Na konci této kapitoly je také krátké pojednání o zkušenostech z prezentace práce na konferenci Excel@FIT.

5.1 Průběh vývoje

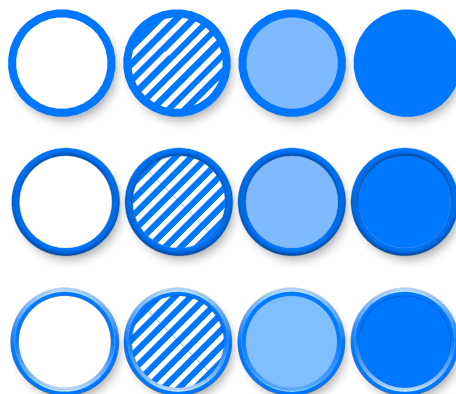
Na počátku vývoje aplikace bylo třeba zprovoznit vývojové prostředí Android Studio (viz kapitola 3.1), seznámit se s ním a připravit nástroje pro ladění aplikace. Pro vývoj aplikace jsem zvolil směr postupu od uživatelského rozhraní přes základní logiku hry až ke konkrétním funkcím. Tento postup umožnil ladit nové části programu už zařazené do uživatelského prostředí, do kterého patří. Tato část práce vychází ze získaných informací (viz kapitola 3) a popisuje realizaci návrhu aplikace (viz kapitola 4).

5.1.1 Prototyp grafického herního prostředí

Jedním z prvních úkolů bylo vytvořit grafiku herní karty, neboť karta je ústředním prvkem celé hry. Grafická komponenta musí být schopna nejen zobrazit kartu a její vlastnosti, ale také reagovat na akce uživatele. Pro ztvárnění karty se nabízelo několik možností. Mohlo se jednat o tlačítko, které na svém pozadí zobrazí kartu a může reagovat na kliknutí (dotyk) od uživatele. Nabízela by se např. komponenta `ImageButton`, která zobrazí obrázek a umí reagovat na stisk. Karty na stisk reagují tím způsobem, že se označí jako vybrané a dalším stiskem je možné jejich výběr zase zrušit. Z tohoto pohledu se také nabízí prvek typu `ToggleButton`, který je uzpůsobený právě pro přepínání stavu.

Třetí možností bylo vytvořit novou vlastní grafickou komponentu založenou na obecném prvku `View` (viz kapitola 3.2.1) a nadefinovat veškerý její vzhled i chování a reakce na akce uživatele. Nakonec byla zvolena tato třetí možnost a byla vytvořena komponenta `CardView`. Tím je umožněno kromě definice vzhledu a chování přesně na míru také zapouzdřit některé operace potřebné pro napojení na herní logiku uvnitř samotné komponenty. Jediné, co je u `CardView` potřeba nastavit pro zobrazení karty je samotný objekt karty (třída `Card`).

Vykreslování grafiky je implementováno v metodě `onDraw`. Opět bylo potřeba zvolit způsob vykreslování obrazců na karty. První ze zkoumaných možností byla možnost vytvořit grafiku jako vektorové obrázky typu SVG v programu Inkscape a obrázky poté importovat



Obrázek 5.1: Jednotlivé návrhy vzhledu obrazců na kartě vytvořené v programu Inkscape. Návrhy jsou zobrazeny v jedné barvě, ale každý návrh je ukázán pro všechny možné typy vyplnění. Pro poloviční vyplnění jsou dokonce dvě možnosti (poloprůhledné a šrafované). První řádek ukazuje prostý styl obrazců, na druhém řádku je aplikován filtr s názvem „Nárožní okraj“ a na třetím filtr „Relief print“.

do Android Studia (viz kapitola 3.1.1). V Android Studiu byl vytvořen základní návrh vzhledu obrazců a několik pokročilých návrhů s použitím různých grafických filtrů, které nabízí program Inkscape. Jednotlivé návrhy ukazuje obrázek 5.1.

Po vytvoření návrhů se vyskytl problém s importem pokročilých filtrů do Android Studia, takže bylo možné importovat jen grafiku bez filtrů, nebo už v Inkscape exportovat obrazce do rastrového formátu a až ty potom importovat do Android Studia. Použití rastrových obrázků jsem se chtěl od začátku vyhnout, jelikož zabírají více paměti a mohou se chovat neočekávaně (např. u nestandardního rozlišení obrazovky). Nakonec tedy byla zvolena možnost použití jednoduché grafiky, což vzhledem k celkovému designu aplikace hodnotím spíše kladně.

Další možnost vykreslování grafiky, kterou jsem se zabýval, bylo vykreslování veškeré grafiky programově. Tato možnost mě od začátku zaujala svou flexibilitou vykreslení, nezávislostí na rozměrech a poměru stran karty. Nevýhodou tohoto přístupu je, že veškerá grafika musí být propočítána a definována v kódu metody `onDraw`. Navíc lze takto vykreslovat pouze jednodušší grafické obrazce, ale to v tomto případě vůbec nevadilo, protože vzhled karet je založen právě na jednoduchých tvarech.

Byla také prozkoumána možnost kombinace přístupů, tedy vykreslení importované grafiky po částech, čímž by se značně omezil nutný počet importovaných souborů. Tento kombinovaný přístup však zdědil spíše nevýhody obou předchozích přístupů. Množství souborů k importu by ani tak nebylo malé a výpočty pro vykreslování by stejně probíhaly přímo v kódu metody. Nakonec jsem se zjistil, že vyjde velmi podobně, jestli na určitá místa a do určité oblasti vykresluji základní tvar nebo obsah vektorového souboru, ve kterém je právě ten základní tvar. Rozhodl jsem se použít čistě programové vykreslování grafiky karet, které mi umožnilo efektivně vykreslit jednoduché tvary i s efektem stínu, což je jediný pokročilejší grafický prvek na kartách.

K rozložení prvků `CardView` do mřížky 3×3 byl použit kontejner `ConstraintLayout`, který umožňuje přidání vodících linek a nastavení pružných podmínek rozložení prvků, které neudávají absolutní hodnoty, ale vztahují se k pozici a velikosti dalších prvků. Do zmíněného kontejneru byly vloženy dvě vertikální a dvě horizontální vodící linky, které jej

v každém směru rozdělují na třetiny. Karty byly roztaženy mezi vodicími linkami a okraji, takže vždy ideálně vyplní příslušnou část obrazovky bez ohledu na rozlišení nebo poměr stran displeje.

Reakce na kliknutí na kartu je definována použitím metody `View.setOnClickListener`. Třída `CardView` obsahuje metody pro přepnutí stavů „vybráno“ a „nevybráno“. Stav „vybráno“ je znázorněn větším vyvýšením a zešednutím pozadí karty. Metody pro přepnutí stavů umožňují nastavit čas přechodu, takže přechod mezi stavy může být pozvolný, čehož je využito při označení karty hráčem. Přechod se také může provést naráz, čehož se využívá při programové manipulaci s kartami (např. sbírání a doplňování karet při uhodnutém setu). Ve třídě `CardView` byly implementovány také metody `hint` a `mistake` pro zvýraznění nápovědy nebo chyby prostřednictvím přechodu do zelené resp. červené barvy a zpět. I těmto metodám je možné nastavit dobu trvání přechodu.

5.1.2 Základy herní logiky

Logiku herní karty implementuje třída `Card`. Vlastnosti karty jsou určeny výčtovými typy, jejichž definice se nachází rovněž ve třídě `Card`. Kromě konstruktoru a přečtení inicializovaných vlastností nabízí třída také metodu `isSet` pro určení, je-li trojice karet platným setem. Metoda zkontroluje každou vlastnost zvlášť. Nastaví čtyři lokální logické proměnné, pro každou vlastnost jednu. Do nich ukládá hodnotu `true`, pokud se vlastnost na všech kartách rovná, nebo pokud se vlastnost na žádné dvojici karet neshoduje, jinak ukládá hodnotu `false`. Výsledná hodnota, kterou metoda vrací, je konjunkcí¹ dílčích hodnot pro jednotlivé vlastnosti.

Vzhledem k tomu, že se tato metoda používá také v cyklech, které prohledávají skupiny karet, jestli v nich není žádný set, může být volána mnohokrát za sebou. Dlouho jsem usiloval o zjednodušení ověření platnosti setu např. pomocí reprezentace karet číslem a nějaké šikovné matematické operace nebo pomocí binární reprezentace a bitových operací, ale toto mé úsilí se nedočkalo úspěchu.

Další metoda třídy `Card` se nazývá `findSet` a umožňuje hledat set ve skupině karet. Hlavní částí metody je vnořený cyklus o třech úrovních, který prochází všechny kombinace tří karet z dané skupiny. V každé iteraci cyklu se kontroluje, jestli trojice karet je nebo není platným setem. V případě úspěchu metoda vrací indexy daných tří karet, jinak hledání pokračuje dál. Pokud cyklus dospěje až na konec, znamená to, že nenašel žádný set a metoda vrací hodnotu `null`. Původně začínala vnější úroveň cyklu od prvního prvku pole. Další úrovně začínaly od dalších prvků a dospěly až ke konci pole. Později byl cyklus změněn tak, že vnější úroveň začíná od třetího prvku a pokračuje na konec pole. Vnořené úrovně pak v každé iteraci prohledávají pole od začátku až k prvku vybranému nejvyšší úrovní cyklu. Tento poněkud zvláštní způsob prohledávání zajišťuje, že budou vybrány vždy karty s co možná nejmenšími indexy. To souvisí se snahou rozdávat pokud možno karty z vrchu balíčku.

Balíček karet byl původně inicializován systematicky postupně podle vlastností karet a karty při rozdávání vybírány z náhodných míst balíčku. Až poté byl zkontrolován výskyt setů v rozdaných kartách a v případě neúspěchu byly právě rozdané karty vráceny zpět do balíčku a celý proces se opakoval. Tento přístup byl později zavržen a změněn. Karty se po počáteční inicializaci zamíchají a při rozdávání nebo doplňování karet se v balíčku hledají karty, které spolu s případnými dříve rozdanými kartami utvoří set. Celý proces rozdávání karet je tak naprosto deterministický a (pseudo)náhoda hraje roli pouze na začátku při mí-

¹Konjunkce je spojení logických hodnot pomocí spojky „and“.

chání karet. Tato modifikace byla nutná pro umožnění inicializace hry pomocí čtyřmístného kódu, ale nový způsob míchání a rozdávání také více odpovídá realitě.

5.1.3 Vysvětlení pravidel a tutoriál

Stránka s vysvětlením pravidel poskytuje nejprve text popisující vlastnosti karet a podmínky platného setu. Aby byl text dobře čitelný a přehledný, bylo třeba ho naformátovat. Za tímto účelem jsem zkoumal možnosti využití HTML a CSS pro naformátování textu a zobrazení na stránce s pravidly. V aplikacích pro Android je možné statickou stránku naformátovat pomocí HTML a CSS a pak ji zobrazit v prvku `WebView`. Text jde formátovat také přímo v definici designu stránky v souboru XML. Tato možnost je však méně přehledná. Text s různým formátem musí být rozdělen do různých komponent designu. Základní formátování (např. tučné písmo a kurzíva) lze nastavit v jiném souboru XML, ve kterém jsou definovány hodnoty řetězců. Této možnosti jsem využil. Díky tomu jsem mohl upustit od nutnosti HTML a CSS a použil jsem formátování v XML definici designu stránky, neboť stačilo rozdělit předformátovaný text na nadpisy a odstavce.

Vysvětlení pravidel je doplněno sedmi příklady trojic karet. Při vkládání příkladů jsem zvažoval přidat je do stránky jako obrázky. Nakonec jsem se rozhodl využít hotové komponenty `CardView` a pouze nastavit přesné požadované hodnoty karet zamýšlených jako příklady. K podání doplňujících informací o každém příkladu jsem zase použil tabulku s vysvětlením, která původně slouží k vysvětlení chyby a k nápovědě v tutoriálu. Příklady jsou voleny tak, aby se navzájem co nejvíc lišily, a tak pokryly co největší šíři možností, které ve hře mohou vzniknout.

K implementaci tutoriálu byla použita již existující stránka s herním prostředím. Tato stránka byla pouze upravena a rozšířena, aby mohla sloužit jak pro klasickou hru, tak pro tutoriál. V několika metodách herního prostředí se nejprve dělí činnosti podle toho, zda se jedná o tutoriál nebo klasickou hru, ale většina operací funguje úplně stejně v obou případech.

Jednou z odlišností tutoriálu je, že když se hráč dopustí chyby, je mu jeho chyba vysvětlena – ukáže se dialog popisující slovně i graficky, které vlastnosti podmínky setu splňují a které je naopak porušují a proč. Dialog s vysvětlením chyby ukazuje obrázek 5.2. Další odlišností tutoriálu je třístupňová nápověda. První stupeň napoví jednu kartu a třetí stupeň celý set. Druhý stupeň nápovědy poradí vlastnosti navrhovaných tří karet. K tomu je použit stejný dialog jako pro vysvětlení chyby.

5.1.4 Ukládání dat

Data, která je třeba uchovat i po vypnutí aplikace, se ukládají do souborů k tomu určených, které se označují jako Shared Preferences (viz kapitola 3.3.2). Nejdůležitější data, která se ukládají, jsou rozehrané hry. Uložení hry probíhá, když se do popředí dostane jiná stránka nebo aplikace. Toho je docíleno implementací metody `onPause`, jejíž spouštění je kontrolováno systémem (viz kapitola 3.3.1). Ukládání a zastavení hry v metodě `onPause` má za následek uložení hry právě v momentu, kdy došlo k přerušení a hra tak bude uložena při každém přerušení. Druhým efektem tohoto ukládání je, že kdykoli je hra přerušena nebo se na popředí dostane jiná aplikace, pozastavuje se měření času. Takto uložená hra je znovu načtena v metodě `onResume`.

O uložené hře je potřeba ukládat následující údaje: uplynulý čas, kód hry a všechny karty ve hře. Uplynulý čas se ukládá jako číselná hodnota v milisekundách. Kód hry se ukládá jako řetězec. Také rozdané karty a karty v balíčku se ukládají jako řetězec, který se

| | | |
|-------|---|-----------------------|
| Barva | ✓ | 3x zelená |
| Počet | ✓ | jeden, dva, tři |
| Výplň | ✗ | 2x poloviční, 1x plná |
| Tvar | ✗ | 2x čtverec, 1x kruh |

OK

Obrázek 5.2: Vysvětlení chybně vybraného setu. Barva na kartách podmínku splnila – byla všude stejná. Počet obrazců splnil druhou podmínku – byl na každé kartě jiný. Výplň a tvar obrazců podmínky nesplnily.

označuje jako stavový řetězec hry. Jednotlivé znaky řetězce reprezentují jednotlivé karty. Prvních devět znaků řetězce reprezentuje rozdané karty a všechny ostatní znaky pak karty v balíčku, a to v přesném pořadí, jak se karty v balíčku vyskytují. Na pořadí karet v balíčku záleží, které karty budou kdy rozdány. Na pořadí již rozdaných karet prakticky nezáleží.

Převod karty na znak probíhá následovně. Každé vlastnosti karty je přiřazena hodnota 0–2. Hodnoty jednotlivých vlastností jsou vynásobeny mocninami 3 a následně sečteny. Můžeme tak dostat čísla karet 0–80. K číslu karty je dále přičtena konstantní hodnota 40 a číslo je převedeno na znak s danou hodnotou. Konstanta 40 se přičítá, protože znaky s ASCII hodnotou 0–31 jsou řídicí znaky, které není dobré v řetězcích používat. Pokud má karta na vstupu převodní metody hodnotu `null`, metoda vrací znak s hodnotou 32 (mezeru). Znaky s hodnotou 33–39 nebo 121–126 nejsou v kódování karet využity.

5.1.5 Nejlepší výsledky

Pokaždé, když hráč dokončí hru, jsou načteny nejlepší dosažené výsledky ze souboru „Shared Preferences“. Už při načítání výsledků je s nimi porovnán nový výsledek a případně je zařazen mezi ně. Pokud se nový výsledek může zařadit mezi nejlepší výsledky, hráč je vyzván k zadání svého jména. Pokud své jméno zadá a dialog potvrdí, jeho výsledek se uloží mezi nejlepší časy a ostatní horší časy se posunou o jednu příčku níže. Hráč může také dialog odmítnout. V takovém případě jeho výsledek vůbec nebude uložen.

Z úvodní nabídky je možné přejít na stránku se zobrazením nejlepších výsledků. Záznamy výsledků jsou tu zobrazeny ve formě tabulky. Při inicializaci stránky je připravena prázdná tabulka. Program pokaždé načítá výsledky ze souboru a postupně je vkládá do tabulky. K tomu je ve zvláštním souboru XML připraven jeden řádek tabulky se čtyřmi textovými buňkami. Podle tohoto souboru se pro každý záznam vytvoří nová komponenta, do příslušných buněk se vloží načtené hodnoty a celý řádek je poté přidán do tabulky. Stránka s přehledem nejlepších výsledků obsahuje také tlačítko umožňující smazat všechny záznamy. Při požadavku na smazání nejlepších výsledků je uživatel vyzván k potvrzení, aby nedošlo ke smazání výsledků omylem.

5.1.6 Založení hry podle kódu

Kód hry je čtyřmístný řetězec, který se skládá ze znaků 0–9 a A–Z (dohromady 36 znaků). Při inicializaci nové hry se kód převede na číslo, jako kdyby reprezentoval číselnou hodnotu v šestatřicítkové soustavě. Toto číslo je poté použito k inicializaci generátoru pseudonáhodných čísel jako parametr „seed“. Generátor pseudonáhodných čísel slouží k zamíchání balíčku. Při inicializaci balíčku se karty sice vytvářejí systematicky postupně, ale jsou přidávány na náhodné místo balíčku. Tím jsou karty v balíčku už od začátku zamíchané. Pokud někdo chce zopakovat konkrétní zamíchání karet, stačí zadat stejný kód hry. Tím se inicializuje generátor pseudonáhodných čísel, a tedy vrací stejnou posloupnost čísel jako v původním případě.

Možnost začít hru podle zadaného kódu byla původně součástí vstupní nabídky, ale po čase byla vyhodnocena jako méně důležitá, a proto byla ukryta za kód hry přímo na herní stránce. Pokud uživatele zajímá, co kód znamená, může na něj kliknout a objeví se mu vysvětlení. Zároveň je mu sděleno, že když kód hry podrží (tzv. „long click“), může začít novou hru s vlastním zadaným kódem. Při tvorbě nové hry s náhodným kódem jsou záměrně vynechány znaky pro písmeno „O“ a číslo „0“ pro jejich zaměnitelnost, ale když uživatel zadává vlastní kód, může zadat všechna čísla i písmena. Pokud uživatel zadá malá písmena, budou automaticky převedena na velká.

5.1.7 Zpětná kompatibilita až k Android 4.1

Od začátku vývoje nebyl na zpětnou kompatibilitu se staršími zařízeními brán zřetel. Minimální požadovaná verze byla zkrátka určena použitými funkcemi. Minimální úroveň API (viz kapitola 3.3.3), která přímo souvisí s verzí systému, byla stanovena na hodnotu 21, jež odpovídá verzi systému Android 5.0. Podle Android Studia aplikace s minimální úrovní API 21 pokryje 71 % uživatelů, což se původně zdálo být v pořádku. Už v počátcích testování však projevilo zájem o aplikaci více lidí se starším systémem verze 4.*, kterým byla aplikace prozatím nepřístupná. Proto byly zkoumány možnosti rozšíření zpětné kompatibility na starší zařízení.

Jako klíčový aspekt omezující zpětnou kompatibilitu se ukázalo použití stínování (viz kapitola 3.2.3), které je dostupné až od úrovně API 21. Vzhledem k faktu, že stínování komponent je prakticky jediný aspekt celé aplikace vyžadující tuto novější verzi, rozhodl jsem se zpětnou kompatibilitu rozšířit. Pro zařízení s verzí Android 5.0 a novější se nic nemění. V programu při práci se stíny byly použity metody zajišťující zpětnou kompatibilitu. Ta je zajištěna tím způsobem, že na starších zařízeních se stíny zkrátka vynechají. Kvůli absenci stínu je na starších zařízeních hůře rozpoznatelný okraj karet. Z toho důvodu byl kartám přidán slabý okrajový rámeček, který i na nových zařízeních lépe zvýraznil okraj. Aplikace je tak dostupná pro všechny systémy Android od verze 4.1, což odpovídá pokrytí přibližně 99,2 % aktivních zařízení.

Kompatibilitu by bylo možné rozšiřovat i dále, ale byly by nutné další zásahy v kódu. Pro rozšíření kompatibility k úrovni API 14 by bylo třeba použít úplně jiný systém metod zajišťujících provedení akce po dokončení animací nebo rozlišovat metody pro různé verze při čekání na nastavení rozložení prvků a několik dalších změn. To by vedlo k rozšíření dostupnosti pro všechny uživatele aktivní na Google Play, ale změny už provedeny nebyly. Vycházel jsem z předpokladu, že 0,8 % uživatel je malá část a navíc se časem ještě bude zmenšovat, proto nemá smysl zavádět do kódu použití starších postupů, které by v budoucnu mohly působit další problémy.

5.2 Ladění, testování a publikování aplikace

Ladění aplikace probíhalo po každé nové změně. Stejně tak ani testování nebylo jednorázovou akcí navazující na ukončení vývoje. První fáze testování, a to jak na různých zařízeních, tak i na uživateli, probíhala už v průběhu vývoje. První testovací fáze pomohla odhalit mnohé nedostatky už v době jejich vzniku. Druhá fáze testování byla spuštěna, když aplikace byla už téměř hotová. Druhá testovací fáze byla systematictější, zaměřená na ověření různých aspektů aplikace. Taktéž publikování aplikace na Google Play bylo průběžným procesem a publikovaná aplikace sloužila už při testování. Každá výraznější změna v aplikaci byla na Google Play publikována jako nová verze aplikace.

5.2.1 Ladění v průběhu vývoje

Android Studio poskytuje velké množství nástrojů k ladění. Umožňuje napojení na fyzická i virtuální zařízení, dále možnost zastavení programu v určitém místě kódu a případné krokování po jednotlivých příkazech. Dovoluje také správu výjimek, jejich zpracování pro následné odhalení opravení příčin. Android Studio umožňuje také správu virtuálních zařízení. Z počátku vývoje byla aplikace laděna na virtuálním zařízení běžícím na emulátoru. Průběh ladění byl často velmi nepříjemný, jelikož jak Android Studio, tak běžící emulátor zabírá značné zdroje počítače, což velmi omezuje výkon. Jednoduché operace, jako je otevření souboru v Android Studiu, pak trvaly třeba i půl minuty. O složitějších úkonech, jako je vyhledávání informací na internetu v průběhu ladění, ani nemluvě.

Podstatné zlepšení průběhu ladění přineslo použití fyzického zařízení. Nejdříve sice bylo velmi náročné zprovoznit vývojářské nástroje na mém telefonu značky Xiaomi, ale když se to povedlo, veškeré ladění probíhalo plynule bez větších omezení. Veškeré funkce používané při ladění zůstaly k dispozici i u použití fyzického zařízení. Android Studio také umožňuje připojit se k již běžícímu procesu, takže pokud jsem na zařízení, kde je nainstalována ladicí verze aplikace dospěl do nějakého neočekávaného stavu, stačilo zařízení připojit k počítači, napojit Android Studio na dříve spuštěný proces a prozkoumat neočekávaný stav podrobněji zevnitř.

5.2.2 Testování na různých zařízeních

Když byla většina klíčových částí aplikace připravena, začalo také průběžné testování. Aplikace byla zpočátku vyzkoušena na náhodných zařízeních, většinou ve vlastnictví členů mé rodiny a přátel. Touto první fází testu byly odhaleny některé problémy například nedostupnost aplikace pro zařízení se staršími verzemi systému. V reakci na toto zjištění byla aplikace upravena, aby byla rozšířena její dostupnost i pro starší zařízení (viz kapitola 5.1.7).

V hlavní fázi testu byla aplikace testována na 12 fyzických zařízeních. Většina z nich byly fakultní přístroje, které jsou studentům dostupné právě pro podobné účely. Zařízení, na kterých byla aplikace testována, jsou uvedena v tabulce 5.1. Cílem testu bylo vyzkoušet, zda se aplikace chová správně na různých zařízeních s různým rozlišením displeje. Na testovacích zařízeních běží různé verze systému Android. Zařízení jsou také od několika různých výrobců, z nichž každý může k systému přidávat vlastní úpravy.

Testovací zařízení bylo připojené k počítači USB kabelem a testování probíhalo následovně:

1. Aplikace byla na zařízení nainstalována.

Tabulka 5.1: Aplikace byla otestována na různých zřízeních od různých výrobců s různými verzemi a adaptacemi systému.

| Č. | Značka | Typ zařízení | Verze systému |
|----|---------|----------------|---------------|
| 1 | HTC | One X | Android 4.2.2 |
| 2 | HTC | One M7 | Android 5.0.2 |
| 3 | Samsung | Galaxy S7 Edge | Android 7.0 |
| 4 | HTC | One M9 | Android 7.0 |
| 5 | LG | Nexus 5X | Android 7.1.1 |
| 6 | Xiaomi | Mi A1 | Android 7.1.2 |
| 7 | Cat | S60 | Android 6.0.1 |
| 8 | LG | Nexus 5 | Android 6.0.1 |
| 9 | Nvidia | Tablet Shield | Android 7.0 |
| 10 | Xiaomi | Redmi 4X | Android 7.1.2 |
| 11 | Samsung | Galaxy S4 Mini | Android 4.4.2 |
| 12 | Samsung | Galaxy S3 Neo | Android 4.4.2 |

2. Bylo vyzkoušeno správné zobrazení všech stránek aplikace a správné chování klíčových funkcí (vybrání setu, rozdání karet, vybrání chybného setu...).

3. Na závěr byly pořízeny snímky obrazovky herního prostředí a vstupní nabídky.

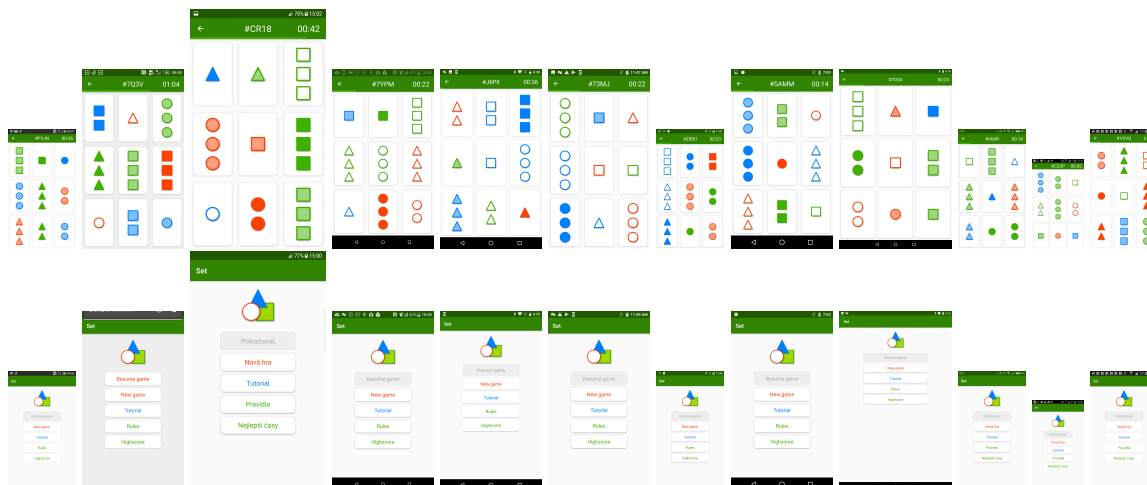
Testování ukázalo, že zobrazení je na různých zařízeních konzistentní a nikde nejsou žádné výraznější odchylky. Shromážděné snímky obrazovky jsou ukázány na obrázku 5.3.

5.2.3 Testování na uživateli

První fáze testování na uživateli probíhala neformálně volným pozorováním. Uživatel byl dáno zařízení se spuštěnou hrou na stránce vstupní nabídky a uživatel byl sledován, co bude dělat a jak bude reagovat. Testéři byli požádáni, aby své myšlenky, dojmy a postřehy sdíleli nahlas pro lepší pochopení. Test byl zaměřen na intuitivnost uživatelského rozhraní a na správné pochopení jednotlivých částí aplikace. Už tato fáze testování odhalila několik základních nedostatků a umožnila zapracovat jejich nápravu.

Jeden z odhalených nedostatků bylo číslo značící počet zbývajících karet v balíčku. Toto číslo bylo nahrazeno nabíhající čarou – „progress barem“. Dále si testéři nevšimli, že přibýlo deset sekund jako penalizace, když zvolili chybný set. Tento nedostatek byl napraven přidáním animace. Kromě červeného probliknutí karet při každé chybě z časomíry „odletí“ červený text „+00:10“ signalizující nárůst času o 10 sekund. Této animace si hráči obvykle poprvé nevšimnou, podruhé ji zaznamenají a potřetí už přesně vědí, kolik času je každá chyba stojí. Na základě poznatků z první fáze uživatelských testů byla také odstraněna možnost vytvoření hry podle vlastního kódu z hlavní nabídky, jelikož byla spíše matoucí. Tato možnost zůstala zachována, jen byla „skryta“ pod text kódu přímo na herní stránce aplikace. První uživatelské testy probíhaly v době, kdy ještě nebyl připraven tutoriál, ale pouze více zdůraznily velkou potřebu tutoriálu k aplikaci přidat. Princip hry je prostý, ale pravidla jsou náročná na pochopení.

Ve druhé fázi uživatelských testů byla aplikace zveřejněna na Google Play a rozšířena převážně prostřednictvím sociálních sítí. Uživatelé byli požádáni o vyplnění dotazníku, který



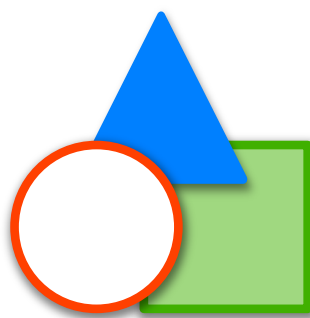
Obrázek 5.3: Porovnání snímků obrazovky klíčových částí aplikace na různých zařízeních ukazuje konzistentnost zobrazení nezávisle na typu zařízení, verzi systému nebo rozlišení obrazovky. Jednotlivé snímky odpovídají záznamům v tabulce 5.1.

se zaměřoval zejména na pochopení pravidel, přehlednost herního prostředí a celkový dojem z používání aplikace. Dotazník vyplnilo 22 testerů, většina z nich byla ve věku 18–30 let. Při instalaci a spuštění aplikace se nevyskytly žádné problémy. Jediná negativní zkušenost v tomto ohledu byla způsobena starou verzí operačního systému (Android 4.0.4) na zařízení jednoho z testerů.

Pravidla hry pochopila většina testerů. Téměř čtvrtina testerů pochopila pravidla hned po jejich přečtení a prohlédnutí příkladů, další pětina testerů pomohl s pochopením jiný člověk a bezmála polovina testerů pomohl s orientací v pravidlech tutoriál. Jeden člověk pravidla nepochopil vůbec a jeden také uvedl, že hru znal už z dřívějšíka. Téměř polovina testerů si prošla všechny úrovně tutoriálu. Další třetina ukončila tutoriál předčasně. Testeři pravidla už pochopili a tutoriál byl pro ně příliš snadný a zdoluhavý. Více než tři čtvrtiny testerů objevily v tutoriálu nápovědu nebo některé její části, někteří z nich nápovědu vůbec nepotřebovali.

Více než tři pětiny testerů hra bavila. Někteří vnímali jako problém, že se občas nemohou posunout dál, protože se jim dlouhou dobu nedaří najít set. Někteří také navrhovali zařazení nápovědy do hry na čas. Osobně jsem se tomuto návrhu dlouho bránil, ale nakonec jsem se přece rozhodl do hry na čas přidat nápovědu za cenu třicetisekundové penalizace. Tím se zajistí, že nápověda nebude nadužívána, ale pomůže hráčům pohnout se dál ze „zamrzlé“ situace.

Tři čtvrtiny testerů si všimli na první pohled nenápadného ukazatele postupu hry („progress baru“), což značí, že i přes počáteční obavy je poměrně dobře viditelný. Naopak význam kódu hry pochopila jen čtvrtina uživatelů. Vysvětlení ke kódu hry bylo mírně vylepšeno, ale žádné další změny v tomto ohledu prováděny nebudou, jelikož se jedná o méně významný aspekt aplikace sloužící pouze opravdovým nadšencům. Desetisekundovou penalizaci za chybný set pochopila většina testerů. Někteří testeři také podotkli, že mají problém s rozpoznáním, která karta je zrovna označená a která ne. V reakci na tento poznatek byla ještě více ztmavena šedá barva pozadí vybrané karty, aby tak vybrané karty lépe kontrastovaly s ostatními nevybranými bílými kartami.



Obrázek 5.4: Logo aplikace použité nejen na Google Play. Je také zobrazeno v hlavní nabídce aplikace, nebo jako ikona aplikace.

Jeden z testerů také uvedl, že má problém s rozeznáváním barev červené a zelené obzvláště u nevyplněných obrazců. Tento jev je pravděpodobně způsoben faktem, že daný člověk trpí daltonismem, tedy poruchou barvocitu neboli částečnou barvoslepostí. Možnosti přizpůsobení aplikace, aby byla použitelná i pro lidi s touto zrakovou vadou, by mohly být předmětem dalšího zkoumání. Řešení tohoto problému není triviální, jelikož hra je na barvách do značné míry založena. Řešením by mohlo být přesně upravit odstíny barev, aby byly rozeznatelné pro většinu lidí trpících různými typy daltonismu nebo alespoň pro některé z nich.

5.2.4 Vydání aplikace na Google Play

Vydat aplikaci na Google Play je možné ve čtyřech stupních – tzv. kanálech (viz kapitola 3.3.4), přičemž první stupeň byl uveden do provozu až v průběhu vývoje, takže vůbec nebyl využit a i kdyby tam byl dřív, pravděpodobně by nebyl potřeba. Jako první byl využit tzv. kanál alfa s uzavřenou skupinou testerů pro testování v průběhu vývoje. První fáze se účastnilo několik dobrovolníků. Vydání aplikace v kanálu beta bylo použito pro druhou fázi uživatelských testů. V závěru vývoje byla aplikace vydána v hlavním produkčním kanálu². V současnosti je aplikace přes Google Play nainstalována na 40³ zařízeních.

Pro vydání aplikace na Google Play je nutné sepsat krátký i delší popis aplikace. Všechny texty v aplikaci jsou dvojjazyčně (anglicky a česky), tedy i popisy aplikace na Google Play jsou napsané oběma jazyky. Bylo také potřeba pořídit několik snímků obrazovky a vytvořit logo aplikace. Vytvořené logo ukazuje obrázek 5.4.

5.3 Účast na konferenci Excel@FIT

Na doporučení vedoucího bakalářské práce jsem se rozhodl přihlásit na konferenci studentských prací Excel@FIT 2018. Jsem si vědom faktu, že tato práce byla na konferenci ve stínu mnohem náročnějších a přínosnějších prací vědeckého charakteru. Nicméně navzdory tomuto faktu mi příprava i účast na konferenci dala cenné zkušenosti a možnost svou aplikaci propagovat ve zcela nové skupině lidí. Počet instalací aplikace se po konferenci zvýšil.

²<https://play.google.com/store/apps/details?id=cz.vutbr.fit.set>

³Údaj je ze dne 11. května 2018

5.3.1 Příprava na konferenci

Pro konferenci bylo třeba připravit některé materiály. Prvním z nich byl článek o práci. Tvorba článku mi poskytla cenné zkušenosti ohledně tvorby publikací. Dalším přínosem, který mi tvorba článku poskytla, bylo uspořádání všech informací a témat týkajících se této práce. Všechny tyto zkušenosti a poznatky jsem poté mohl využít i při tvorbě této zprávy.

Dalším požadavkem bylo vytvořit plakát, shrnující slide a propagační video o své práci. To jsou materiály s omezeným prostorem – video má limitovaný čas a plakát i shrnující slide definovanou plochu. Tento fakt mi pomohl zamyslet se a uvědomit si, které aspekty mé práce jsou ty nejdůležitější a které z nich stojí za zmínku. Dále pak tvorba propagačního videa i plakátu jsou v zadání této práce. Materiály z konference jsem mohl převzít a pouze s drobnými úpravami použít zde. Materiály jsou k nalezení na přiloženém CD. Posledním požadavkem konference bylo vytvoření náhledu práce, což je menší čtvercový obrázek typu loga. Jako náhled práce bylo využito už hotové logo (obrázek 5.4) připravené dříve pro Google Play.

5.3.2 Průběh konference

Jelikož tato práce neobsahuje žádný převratný vědecký objev, který bych mohl na konferenci prezentovat, pojal jsem účast na konferenci spíše ve smyslu propagace. Snažil jsem se aplikaci lidem přiblížit, nalákat je k vyzkoušení či nainstalování aplikace. Spoustu lidí hra zaujala, což mě osobně velmi potěšilo. Po dni konference se na Google Play zvýšil počet instalací aplikace. Někteří lidé si samozřejmě aplikaci vyzkoušeli a poté ji zase odinstalovali, ale někteří ji mají stále. Stejně jako u testerů, tak i u lidí na konferenci se ukázalo, že každý chápe pravidla jinak rychle. Některým lidem šlo hraní téměř samo a jiní si nad rozdanými kartami dlouho lámali hlavy, ale i to považuji za pozitivní jev. Kdyby někoho aplikace nebavila, vypnul by ji rovnou a pokračoval k zajímavějším pracím.

Kapitola 6

Závěr

Cílem práce bylo vytvořit aplikaci ztvárňující logickou karetní hru Set pro chytré telefony a podobná zařízení se systémem Android. Aplikace byla úspěšně dokončena a je volně k dispozici na webovém obchodě Google Play¹.

Nejprve bylo třeba nastudovat možnosti vývoje aplikací pro Android. Pro tvorbu aplikace bylo zvoleno prostředí Android Studio. Dále byl proveden průzkum podobných existujících aplikací a vyhodnocení jak jejich kladných stránek, ze kterých je možno vycházet, tak i těch záporných, kterým je dobré se vyhnout. Po úvodní přípravě byl vytvořen návrh a později i samotná aplikace, která byla už v průběhu vývoje testována na různých virtuálních i fyzických zařízeních a také na samotných uživateli. V závěru práce byla aplikace publikována a pro účely propagace vytvořen plakát a krátké video.

Dalším cílem práce bylo rozšířit povědomí o hře Set a usnadnit uživatelům pochopení pravidel. Aplikace je v současné době nainstalována na desítkách zařízení. 73 % testerů uvedlo, že s pomocí různých částí aplikace pochopili pravidla hry Set a 68 % testerů hraní hry bavilo. Aplikace je dostupná pro 99,2 % zařízení aktivních na Google Play

I po odevzdání práce bych chtěl pokračovat v propagaci aplikace, aby se rozšířila mezi více uživatelů. Do budoucna bych se rád pokusil vytvořit k aplikaci mód pro více hráčů. Rád bych, aby hráči mohli propojit více zařízení a hru hrát společně. K propojení zařízení by se mohlo přistupovat buďto přes internet, nebo i lokálně bez nutnosti připojení k internetu. Další prostor pro výzkum by mohl být, jaké zvolit odstíny barev nebo jak upravit vlastnosti karet, aby hra byla přístupná i pro osoby s poruchou barvocitu. V neposlední řadě se nabízí prostor pro výzkum náročnosti jednotlivých setů a následnou optimalizaci tutoriálu.

Jsem rád, že se mi podařilo práci dokončit a zveřejnit hotovou aplikaci. Z práce si odnáším spousty zkušeností s vývojem aplikací pro Android, případně mobilních aplikací obecně. Získal jsem také cenné zkušenosti v oblasti návrhu grafického uživatelského rozhraní a uživatelského testování aplikace.

¹<https://play.google.com/store/apps/details?id=cz.vutbr.fit.set>

Literatura

- [1] Custom View Components. Android Developers.
URL <https://developer.android.com/guide/topics/ui/custom-components>
- [2] Save key-value data. Android Developers.
URL <https://developer.android.com/training/data-storage/shared-preferences>
- [3] Set (game). Wikipedia, the free encyclopedia.
URL [https://en.wikipedia.org/wiki/Set_\(game\)](https://en.wikipedia.org/wiki/Set_(game))
- [4] String resources. Android Developers.
URL <https://developer.android.com/guide/topics/resources/string-resource>
- [5] Understand the Activity Lifecycle. Android Developers.
URL <https://developer.android.com/guide/components/activities/activity-lifecycle>
- [6] What is API Level?: <uses-sdk>. Android Developers.
URL <https://developer.android.com/guide/topics/manifest/uses-sdk-element#ApiLevels>
- [7] DiMarzio, J. F.: *Beginning Android Programming with Android Studio*. New York: John Wiley & Sons, Incorporated, Čtvrté vydání, 2016, ISBN 9781118705599.
- [8] Klarreich, E.: A Simple Proof From the Pattern-Matching Card Game Set Stuns Mathematicians. 2016.
URL <https://www.wired.com/2016/06/simple-proof-card-game-set-stuns-mathematicians/>
- [9] Krug, S.: *Don't Make Me Think, Revisited: A Common Sense Approach to Web Usability*. Voices That Matter, New Riders, třetí vydání, 2013, ISBN 978-0321965516.
- [10] Lacko, L.: *Vývoj aplikací pro Android*. Brno: Computer Press, první vydání, 2015, ISBN 978-80-251-4347-6.
- [11] Phillips, B.: *Android programming : the Big nerd ranch guide*. Atlanta, GA: Big nerd ranch, third edition vydání, 2017, ISBN 978-0-13470-605-4.
- [12] Weinschenk, S. M.: *100 věcí, které by měl každý designér vědět o lidech*. Computer Press, 2012, ISBN 978-80-251-3649-2.